Вінницький національний технічний університет
(повне найменування вищого навчального закладу)
Факультет інформаційних технологій і комп'ютерної інженерії
(повне найменування інституту, назва факультету (відділення))
Кафедра обчислювальної техніки
(повна назва кафедри (предметної, циклової комісії))

Пояснювальна записка
до магістерської кваліфікаційної роботи
магістр
(освітньо-кваліфікаційний рівень)

на тему: Метод балансування трафіку мережі Інтернет сервіс-провайдера з урахуванням вимог до якості обслуговування

Виконав: студент 2 курсу групи 1КІ-18м
спеціальності
123 «Комп'ютерна інженерія»
(шифр і назва напряму підготовки)
Грабовський Є.В
(прізвище та ініціали)

Керівник: к.т.н., доц. Захарченко С.М.
(прізвище та ініціали)

Рецензент: к.т.н, доц.Карпінець В.В
(прізвище та ініціали)

Вінниця ВНТУ – 2019 року

АНОТАЦІЯ

Дослідження присвячено актуальній темі – методи балансування трафіку мережі інтернет-сервіс провайдера з урахуванням вимог до якості обслуговування.

Головні пункти новизни в роботі – новий підхід до балансування трафіку в мережі інтернет-сервіс провайдера , в якому фільтрація трафіку для подальшого поділу його на групи по пріоритетах відбувається за допомогою технології глибокого аналізу трафіку. Самей такий підхід дозволяє використати всі переваги глибокого аналізу трафіку, такі як: збір статистики для маркетинга, захист від зовнішніх атак зловмисників. А використання технології MPLS дозволяє зменшити час за який трафіку буде проходити по мережі інтернет сервіс провайдера, що нівелює затримку, яка виникає при аналізі трафіку.

ABSTRACT

The study focuses on the topical topic - methods of balancing the network traffic of an Internet service provider with regard to the quality of service requirements.

The main points of novelty in the work - a new approach to balancing traffic in the Internet service provider's network, in which the filtering of traffic to further divide it into priority groups occurs by means of technology of deep traffic analysis. This approach allows you to take full advantage of in-depth traffic analysis, such as: collecting statistics for marketing, protecting against external attacks by attackers. And the use of MPLS technology allows to reduce the time for which traffic will pass through the Internet service provider, which eliminates the delay that occurs in the analysis of traffic.

Оглавление

					08-23.МКР.004.00.000 ПЗ		
Змн.	Лист	№ докум.	Підпис				
Розроб.		Грабовський Є.В..			Методи балансування трафіку мережі інтернет-сервіс провайдера з урахуванням вимог до якості обслуговування	Літ.	Арк. 6 / Аркушів 122
Перевір.		Захарченко С.М.					
Реценз.		Карпінець В.В.				ВНТУ, гр. 1КІ-18М	
Н. Контр.		Швець С.І.					
Затверд.		Мартинюк Т.Б.					

Вступ

Актуальних науковим завданням у галузі телекомунікацій є передававання найрізноманітнішого трафіку з дотриманням вимог до якості обслуговування. Це пов'язано з тим, що багато потоків даних передається через мережу, ресурси якої потрібно розподілити між цими потоками. Дані, які підлягають передачі різні за своєю важливістю та природою, то необхідно мати механізми, які допоможуть розв'язувати задачу розподілу ресурсів у відповідності до властивостей тих потоків, що передаються у конкретний момент через конкретні телекомунікаційні вузли. Для підвищення якості обслуговування (QoS) актуальним є пошук методів управління мережевими ресурсами для забезпечення їхнього збалансованого завантаження й гарантованої якості обслуговування різнорідного трафіку користувачів у мультисервісних мережах.

Зв’язок роботи з науковими програмами, планами, темами. Магістерська робота виконана відповідно до напрямку наукових досліджень кафедри обчислювальної техніки в галузі комп’ютерних систем та мереж, а також, спеціальності 123 – комп’ютерна інженерія.

Метою роботи є вдосконалення мережі інтернет-сервіс провайдера за рахунок балансування трафіку з урахуванням вимог до якості обслуговування. Для досягнення поставленої мети необхідно розв’язати такі питання:

— аналіз сучасних рішень по забезпеченню гарантованої якості обслуговування в мультисервісних телекомунікаційних мережах;

— розглянути методи аналізу трафіку;

— розробити теоретичну модель модернізації мережі інтернет-сервіс провайдера з урахуванням до вимог якості обслуговування;

— виконати модернізацію мережі інтернет-сервіс провайдера на основі теоретичної моделі.

Об’єкт дослідження – це сучасні процеси балансування та аналізу трафіку.

Предмет дослідження – це технології балансування та аналізу трафіку.

Методи дослідження. Дослідження, виконані під час роботи над кваліфікаційною магістерською роботою, ґрунтуються на основних поняттях технологій проектування комп’ютерних мереж; багато протокольній комутації по мітках; глибокому аналізу пакетів; середньому аналізу пакетів; поверхневому аналізу пакетів.

Наукова новизна одержаних результатів полягає в такому:

Практичне значення одержаних результатів полягає у такому:

1) Удосконалено взаємодію технології глибокого аналізу трафіку з методами балансування трафіку.

2) Впроваджено підхід до модернізації великих комерційних мереж методом балансування трафіку з урахуванням вимог до якості обслуговування.

3) Удосконалено технологію проектування мереж інтернет-сервіс провайдерів.

4) Розроблено нову концепцію використання глибокого аналізу трафіку.

Отримані результати можуть бути впроваджені в процесі модернізації або проектування корпоративної комп’ютерної мережі інтернет-сервіс провайдера .

Достовірність теоретичних положень магістерської кваліфікаційної роботи підтверджується строгістю постановки задач, коректним застосуванням інформаційних технологій під час доведення наукових положень, строгим виведенням аналітичних співвідношень, збіжністю результатів дослідження з результатами, що отримані під час впровадження розробленої комплексної теоретичної моделі.

Особистий внесок магістранта. Усі результати, наведені у магістерській кваліфікаційній роботі, отримані самостійно і опубліковані у настуному виданні — Технології побудови мережі інтернет-сервіс провайдера / Грабовський Є.В. // Збірник Матеріалів XLVII Науково-технічної конференції факультету інформаційних технологій та комп'ютерної інженерії (2018). Режим доступу: https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2018/paper/view/ 4973/4065.

1 Основні характеристики технології MPLS

1.1 Передісторія виникнення технології MPLS

Технологія багато протокольної комутації за мітках MPLS з'явилася як результат кылькох кілька подібних технології, які були винайдені в середині 1990-тих роках. Ці механізми мають ряд спільних характеристик. Всі вони використовують для відправлення пакетів, простий метод заміни міток і розроблену під Інтернет структуру управління, тобто IP-адреси та стандартні протоколи маршрутизації, наприклад OSPF і BGP.

Зростаюча зацікавленість до комутації з використанням міток призвів до створення спеціальної робочої групи IETF, ціллю якої було виробити на основі вищезазначених механізмів основний стандарт. Щоб не допустити давати групі назву, яка відповідала б продукту якоїсь із компаній, IETF зупинив свій вибір на нейтральній назві: багато протокольна комутація за мітках. Багато протокольна комутація по мітках MPLS називається так тому, що її засоби можуть бути застосовані до будь-якого протоколу мережевого рівня, тобто MPLS - це інкапсулюючий протокол, здатний транспортувати інформацію багатьох протоколів нижчих рівнів моделі OSI.

Фізичний рівень містить функції, що забезпечують використання фізичного середовища для двонаправленої передачі бітів (з такою достовірністю, яку забезпечує це середовище) по прямому напрямку, що зв’язує два вузли мережі. Другий рівень - це рівень ланки даних (канальний) - містить функції, що забезпечує формування в цьому трактуванні надійної логічної ланки зв'язку, в якому проходить двонаправлений обмін інформаційних блоків між необхідними вузлами; при цьому шляхом виявлення і виправлення помилок гарантується задана достовірність пересилання. Третій, мережевий рівень містить функції, що гарантують транспортування інформаційних блоків від відправника до одержувача через декілька вузлів мережі по не характерному маршруту, який складається з вузлів другого рівня. Загальна концепція протоколів всіх рівнів (крім фізичного) полягає в тому, що інформаційний блок данних кожного рівня містить заголовок і

інформаційні поле, і в тому, що блок протоколу вищого рівня переміщається в інформаційне поле блоку протоколу нижчого рівня.

Проте у всіх цих попередніх MPLS операціях не піддавався сумніву базовий принцип: маршрутизатори виконують операції маршрутизації, а комутатори виконують операції комутації, і влаштування цих двох типів завжди функціонують окремо. Маються на увазі не лише IP-маршрутизатори і ATM- комутатори, а й саме правило поділу функцій рівнів 2 і 3 між різними технологіями і пристроями. Подальша історія відповідає вислову, який приписують Альберту Ейнштейну: «Всі давно знають, що те і те абсолютно неможливе. Але ось знайдеться невіглас, який цього не знає, він і зробить відкриття ».

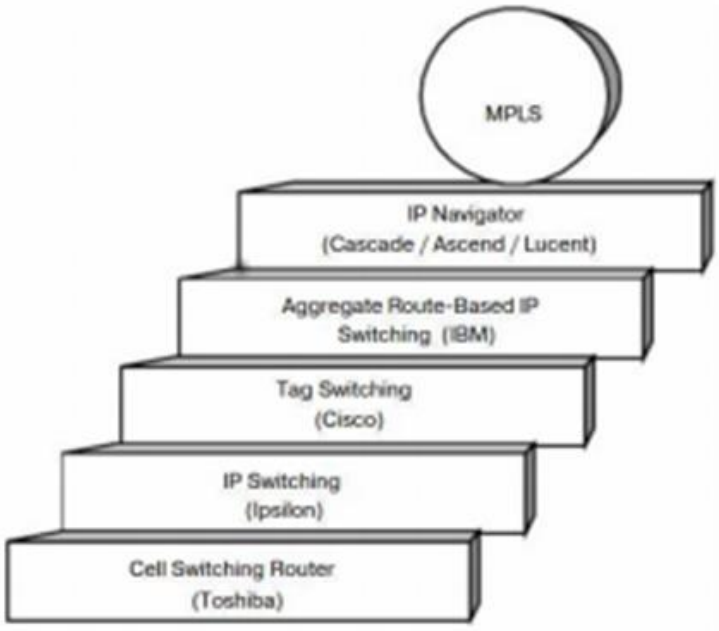

Рисунок 1.1 Етапи еволюції технологій до технології MPLS

У ролі такого «невігласа» виступила компанія Toshiba, яка вперше поставила під сумнів цей принцип і в 1994 році анонсувала маршрутизатор комутації ланок КСВ (стільниковий маршрутизатор комутації). В архітектурі CSR реалізована функція управління комутаційного поля ATM-комутатором за допомогою протоколу IP, а не протоколами сигналізації мережі ATM типу Q.2932. Подібний підхід, зміг би звести нанівець необхідність використання практично всієї сигналізації ATM і всі функції перенаправлення між IP і ATM.

Компанії Ipsilon, завдяки більш повній технічній специфікації IP Switching, наявність готового продукту IP switch, зазвичай приписують створення першої, по-справжньому формалізованої концепції технології MPLS, подобної до комутації міток в мережах IP, що дістала значно більше визнання, ніж технологія КСА. Сам IP-коммутатор складався з ATM-комутатора і контролера IP-коммутатора, який займався функцією управління. Контролер IP-коммутатора фактично був відокремленим пристроєм, що містить функціональні характеристики маршрутизації і пересилання даних в мережі.

Незабаром компанія Cisco Systems показала свій варіант технології комутації по мітках під назвою комутація за тегами (Teг Switching), яка істотно відрізнялася від двох розглянутих вище технологій IP Switching і CSR. Для створення таблиць пересилання в комутаторі вона не грунтувалася на потік трафіку даних і була специфікована для ряду технологій рівня 2, відмінних від ATM. Тому, технологія Tag Switching виявилася кращою для остаточної концепції MPLS, ніж механізм IP Switching. До того ж, технологія MPLS в значній мірі вийшла з механізму Tag Switching. Так званий тег, тобто фіксована кількість біт, що використовується для адресації, багато в чому подібний мітка MPLS. Порядок Tag Switching призначався для спільної роботи з рядом протоколів перших рівнів і вміщував в себе протокол розподілу ключових слів (TDP, Протокол розподілу тегів), як і в MPLS. Нові маршрутизатори могли обслуговувати виклики різних залежностей від необхідної якість обслуговування. До того ж, всі маршрутизатори виробництва Cisco Systems , в яких був реалізований механізм Tag Switching, пізніше були модернізовані та змогли підтримувати технологію MPLS.

Практично відразу ж після того, як Cisco Systems опублікувала інформацію про технологію Tag Switching і повідомила про її стандартизації в IETF, від компанії IBM надійшов проект Інтернет стандарту, в якому пропонувалося інша технологія комутації за мітках — Agregate Route-based IP Switching (APIS). Механізм ARIS призначався для використання в ATM- та FR-коммутаторах, а також в пристроях комутації на рівні 2 в локальних мережах. Пізніше був спроектований пристрій, в якому був реалізований механізм ARIS, який отримав

назву ARIS вбудований комутатор-маршрутизатор (ISR), технологія ARIS має на багато більше спільних характеристик з технологією Tag Switching, ніж з іншими вже раніше згадуваними технологіями, - в обох для створення таблиці маршрутизації використовується трафік інформації керування, а не трафік даних, - але разом з цим технологія ARIS має деякі відмінності від Tag Switching. Основна розбіжність полягає в тому, що ARIS заснований на маршрутах, а не на потоках як Tag Switching. Маршрутизація в домені ARIS будується на базі вихідного вузла. Конфігурується домен ARIS вихідні вузлів, а потім від них поширюється маршрути в сторони вхідних вузлів . Вихідний вузол може бути заданий переліком ідентифікаторів: префіксом одержувача протоколу IPv4, IP-адресою вихідного маршрутизатору, ідентифікатором маршрутизатора OSPF або ідентифікатором пари багатоадресної передачі даних. Маршрути встановлюють незалежно від потоків пакетів.

Ще однією технологією, що була перед MPLS, є технологія IP Navigator, запропонована компанією Cascade. Cascade була пізніше куплена компанією Ascend, яка потім стала частиною компанії Lucent Technologies. В технології IP Navigator було використано багато ідей комутації в IP-мережах, розроблених раніше компаніями Toshiba, Ipsilon, Cisco і IBM.

1.2. Основні поняття технології MPLS

MPLS може розглядатися як сукупність технологій які працюють спільно і забезпечують доставку пакетів від відправника до одержувача в контрольований і ефективний спосіб. У MPLS для пересилання пакетів використовуються комутовані за мітками тракти LSP, які організовані за допомогою протоколів маршрутизації і сигналізації на рівні 3 [1].

Label — мітка - короткий ідентифікатор фіксованої довжини, який визначає приналежність пакета того чи іншого FEC.

FEC - Forwarding Equivalence Class - клас еквівалентності пересилання - безлічі пакетів, що пересилаються однотипно, наприклад, з метою забезпечити заданий рівень QoS.

LER - (MPLS edge router - граничний вузол мережі MPLS) - прикордонний вузол MPLS, що з'єднує домен MPLS з вузлом, що знаходиться за межами доменом.

Label swapping - заміна міток - заміна міток прийнятого вузлом мережі MPLS пакета новими мітками, пов'язаною з тим же FEC, при пересиланні пакета до нижнього вузлу.

Loop detection - виявлення закільцьованих маршрутів мережі - метод, що дозволяє виявити, що пакет пройшов через вузол один і більше раз.

LSP - (Label Switched Path) комутований по мітках тракт — який приходить через один або більше LSR тракт, по якому йдуть пакети одного і того ж самого FEC.

Loop prevention - попередження створення закільцьованих маршрутів в мережі - метод виявлення та усунення закільцьованих маршрутівв мережі.

IPLSR - (Label Switching Router) - маршрутизатор комутації за мітками - маршрутизатор, здатний пересилати пакети за технологією MPLS.

ER - LSP - (explicitly routed LSP) - LSP з явно заданим маршрутом - тракт LSP, який організований способом, відмінним від традиційної маршрутизації пакетів.

MPLS egress node - вихідний вузол MPLS - останній MPLS - вузол в LSP, що направляє вихідний пакет до адреси, який знаходиться поза MPLS-зоною.

MPLS domain - домен MPLS - сукупність вузлів MPLS, між якими існують безперервні LSP.

MPLS ingress node - вхідний вузол мережі MPLS - перший MPLS-вузол в LSP, що приймає вихідний пакет і поміщає в нього мітку MPLS.

1.3 Комутація по міткам

Саме визначення MPLS вказує, що мітки - основа цієї технології. Саме з мітками виконуються всі процедури їх розподілу по маршрутизаторам LSR і процедури створення шляху LSP, за яким будуть слідувати пакети MPLS[2]. Після розподілу міток і створення шляхів LSP може виконуватися основна функція MPLS - пересилання забезпечених мітками пакетів в мережі MPLS. Крім цієї функції

повинні вирішуватися і допоміжне завдання, пов'язане з мітками, а саме, контроль часу збереження міток, впорядкування міток та обробка помилок.

У вказаному прикладі домен мережі MPLS вказаний як сукупність вузлів LSR, між якими створено безперервні LSP з'єднання. Розглянемо докладніше основні кроки алгоритму, що виконується в відношенні пакетів даних в ньому. На базі рис. 1.2, наведені основні кроки, які необхідні, щоб забезпечити проходження пакетів даних через домен MLPS:

Створення та розподіл міток. До початку передачі через мережу MPLS пакет трафіку будь-якого виду маршрутизатор LSR встановлює відповідність між мітками та FEC в своїй таблиці. Далі буде показано, як маршрутизатори, які стоять нижче, за допомогою сигналізації LDP, що використовує транспортний протокол TCP, виробляють розподіл міток і прив'язку їх до класів FEC. Проводиться погодження характеристик трафіку та функціональних можливостей MPLS. Нагадаємо, що значення міток може вибиратися та розсилатися або заздалегідь, тобто до передачі даних (крива 2 на рис. 1.2), або генеруватися як пакети, які належать вступнику в мережу MPLS певного потоку даних чи трафіку певного класу (крива 3). Ці два підходи по призначенню міток, як зазначалося в попередньому розділі, називаються, призначенням з керуванням від програми та призначенням, керованим трафіком (даними) відповідно. Але після того як домен комутації за мітках налаштований для обслуговування пакетного трафіку, який пересилається через мережу MPLS за допомогою міток, тоді всі пакети обробляються однаково.

Створення таблиці в кожному LSR. При отриманні даних про прикріплення міток до FEC кожний маршрутизатор LSR створює записи в таблиці LIB. Як наслідок вміст таблиці відображає відповідність між мітками і FEC та ставить у відповідність кожній потрібній парі «вхідний інтерфейс,в який входить мітка» пару «вихідний інтерфейс,через який виходить мітка». При новому узгодженні прив'язки мітки до FEC записи в таблиці оновлюються. Слід нагадати, що таблиці міток, відповідно до яких кожен пакет направляється за відповідним шляхом LSP, завжди мають бути задані до того, як пакет розпочне свій шлях по мережі. Окрім

того, комутований за мітками шлях - завжди односторонній. Якщо потрібно, щоб пакетний трафік між двома LSR комутаторами проходив і в протилежному напрямку, слід створити два шляхи.

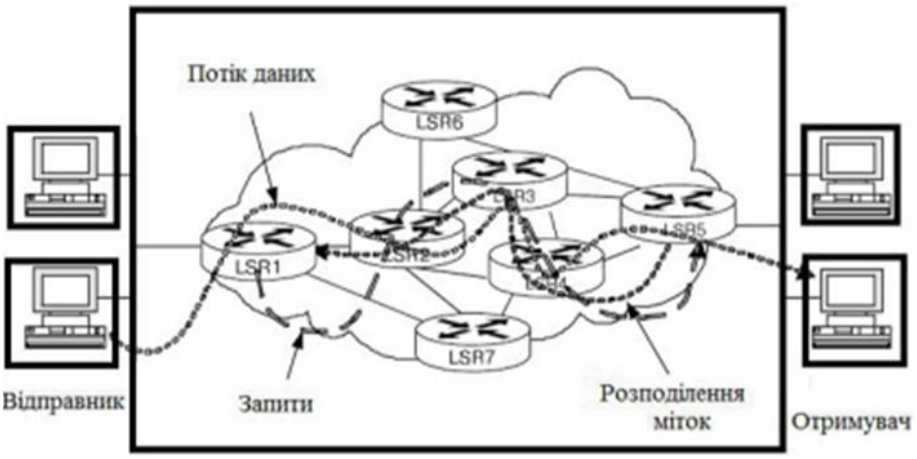

Рисунок 1.2 - Проходження пакету через MPLS домен

Створення комутованого по мітках тракту LSP. Показано лінією 3 на рис. 1.2, тракти LSP створюються в напрямку, який протилежний напрямку створення записів в таблицях LIB. Також важливо, що кожен LSR отримує мітки від нижчого по рівню маршрутизатору. LSP створюється методом послідовної маршрутизації по ділянках мережі, а якщо потрібна оптимізація розподілу трафіку, щоб визначити шлях використовується протокол CR-LDP, що забезпечує виконання вимог до QoS / CoS, або протокол RSVP-TE.

Табличний пошук та інкапсуляція мітки в пакет. Вхідний маршрутизатор (LSR1 на рис. 1.2), визначивши, якому FEC належить прийнятий ним ззовні пакет, використовує таблиці LIB, щоб відшукати потрібну прив'язку «FEC-метка», та інкапсулює цю мітку способом, відповідним застосовуваної на другому рівні технології, як це буде показано нижче.

Пересилання пакета. Розглянемо проходження пакета який йде від вхідного маршрутизатора LSR1 до вихідного маршрутизатора LSR5. Важливо, що LSR1 може не мати мітки для цього пакета. Тоді він знаходить наступний маршрутизатор по IP-адресі. Нехай цим маршрутизатором для LSR1 є LSR2. Тоді маршрутизатор LSR1 ініціює запит мітки від LSR2 для цього пакету даних. Отриману мітку LSR1

вставляє в пакет та пересилає його до LSR2. Кожен наступний LSR аналізує мітку, що міститься в прийнятому пакеті, потім замінює її вихідною міткою і пересилає пакет далі. Коли пакет досягає останнього LSR, відкидає мітку з пакета, оскільки пакет виходить за межі домена MPLS, і доставляє отримувачу. Шлях LSP, за яком проходить пакет, показаний переривчастими лініями 3.

Розглянемо детальніше використовуваний на кроці 5 алгоритм підміни міток при пересиланні пакетів через MPLS домен. Отримавши пакет, маршрутизатор LSR дістає з нього мітку та використовує її в якості індексу в своїй таблиці маршрутизації. Щойно знайдено запис, в якому значення вхідної мітки дорівнює значенню мітки, яку дістали з пакета, маршрутизатор, згідно з підзаписом цього запису, замінює вхідну мітку в цьому пакеті вихідною міткою та пересилає пакет через вихідний інтерфейс, вказаний в підзаписі, до наступного LSR , також зазначеному в цьому підзаписі. У випадку, коли підзапис вказує на певну вихідну чергу, маршрутизатор перенаправляє пакет саме в цю чергу. Простота алгоритму пересилання пакетів, який використовується в MPLS, обумовлює просту та економічну його реалізацію в реалізації апаратного забезпечення, що, дозволяє підвищити продуктивність пересилання даних без використання дорогої апаратури. Якщо LSR підтримує кілька таблиць (по одній для кожного інтерфейсу), то єдина зміна алгоритму заключається в тому, що після отримання пакету LSR попередньо вибирає ту таблицю, яка буде використовуватися для обробки пакета. Вибір таблиці проводиться згідно з ідентифікатором інтерфейсу, через який пакет був отриманий.

Тому, мітка, що переноситься в складі пакету, завжди пересилає семантику пересилання, тому що вона однозначно визначає потрібний запис в таблицях, які веде LSR, і через те що цей запис містить інформацію про те, куди пересилати пакет. В вигляді опції мітка може передавати семантику резервування ресурсів, оскільки обумовлений цим запис може містити інформацію , які ресурси буде використовувати пакет, наприклад, яку вихідну чергу, в яку він повинен направляти. Коли мітка переноситься в заголовку ATM чи Frame Relay, вона має передавати семантику як пересилання та резервування ресурсів. Коли мітка

переходить в спеціальному заголовку, інформація про те, які саме ресурси будуть доступні пакетам, може кодуватися як розділ цього заголовка, а не переписуватися міткою, яка в цьому випадку служить тільки для пересилання.

1.4 Структура мітки

Мітка є коротким елементом фіксованої довжини, що використовується для місцевої ідентифікації класу еквівалентності пересилання FEC. Через значення мітки пакета в кожному вузлі маршруту мережі, по якому він передається, визначається приналежність певного FEC до нього.

Структура мітки MPLS представлена на рис. 1.3. Її довжина становить 32 біта : 12 бітів - заголовок та 20 бітів - значення мітки. Заголовок мітки складається з 3-х полів: 3-бітового поля Exp, яке служить для позначення класу обслуговування, S-біта ознаки досягнення «дна» стека та 8-бітового поля TTL .

20-бітове поле містить значення MPLS-мітки, що може бути числом в діапазоні від 0 до 1048575, виключенням є резервні значення (0, 1, 2, 3 та ін.), визначенням використовування яких займається робоча група MPLS.

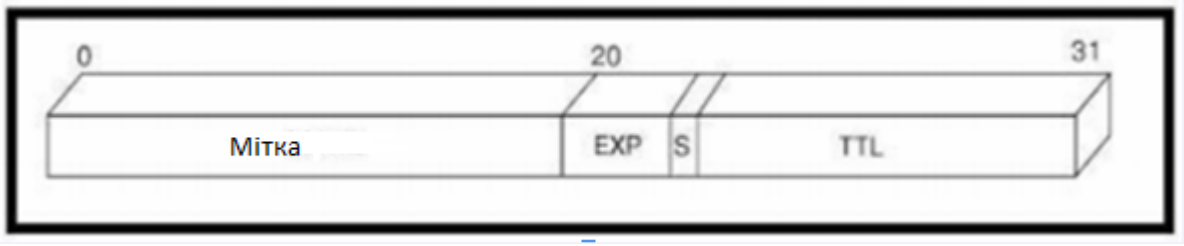

Рисунок 1.3 - Структура мітки пакету MPLS

Розглянемо вміст полів заголовка мітки. Поле експериментальних бітів містить три біта,що зарезервовані для подальших досліджень і експериментів. В даний час проводиться дослідження, спрямовані на створення стандарту використання цих бітів для диференційованого обслуговування різнотипного трафіку та ідентифікації класу обслуговування. Спочатку поле мало назву - «Клас обслуговування» (CoS), і ця назва досі зустрічається в літературі. При наданні диференційованих послуг MPLS це поле може вказувати клас обслуговування, прикладом може слугувати аналогічний класам DiffSe. Для забезпечення наскрізної якість IP-послуг, на прикордоному комутаторі MPLS-мережі дозволяється

скопіювати в поле CoS поле IP-пріоритету з урахуванням того, що поле CoS в заголовку містить 3 біта, і в ньому може передаватися лише 3-бітовое поле IP-пріоритету. При необхідності інформація CoS може передаватися в вигляді однієї з міток MPLS-стеку, тому поле мітки має розмір 20 бітів та здатне вмістити як поле IP-пріоритету, так і поле DSCP.

Біт S це засіб підтримки ієрархічної структури стеку міток MPLS. У заголовку крайньої (тобто найглибшої або нижньої) мітки біт S = 1, а в усіх інших мітках в стеці біт S = 0. Поле часу життя (TTL) в заголовку MPLS аналогічне полю TTL в IP-дейтаграмі; це поле слугує механізмом,який запобігає можливості нескінченної циркуляції пакетів по мережі через утворення закільцьованих маршрутів. Байт TTL знаходиться в самому кінці заголовка мітки і, як представлено на рис. 1.3, займає біти 24 - 31. Діапазон можливих значень цього поля становить від 0 до 255 . Пояснимо його призначення докладніше.

Як згадувалося, протокол MPLS має два способи визначення маршруту, яким слідуватиме пакет. Перший з них аналогічний способу hop-by-hop,який використовується сьогодні в IP-мережах, та передбачає, що кожний маршрутизатор самостійно вибирає, куди пересилати прийнятий ним пакет, так маршрут, по якому транспортується пакет, виявляється умовно випадковим. Другий спосіб за основу взяв те, що маршрутизатори по шляху проходження пакету приймають рішення не самостійно, а відповідно з інструкціями, які отримані від одного з LSR шляху LSP. Таким чином, маршрут проходження пакетів однозначно визначений заздалегідь.

Перший спосіб має певні переваги, пов'язані з порівняною простотою його реалізації, проте не виключає феномен створення маршрутних петель, для боротьби з яким використовується поле TTL. Перший LSR поміщає в це поле максимальну кількість LSR, через які може пройти пакет, в свою чергу кожний маршрутизатор, через який пакет проходить, інкрементує це число на одиницю. Може бути, що пакет не призначений вузлу, де він перебуває в той момент, коли поле TTL прийняло нульове значення, цей пакет відкидається.

Значення TTL, встановлене на кордоні MPLS-мережі, зменшується після проходження пакетом кожного наступного LSR. При введенні в пакет мітки поле

TTL протоколу IP копіюється в поле TTL MPLS. Що надає утиліті traceroute можливість показувати всі проміжні вузли мережі MPLS в випадку, коли при досягненні точки призначення пакет проходить через домен MPLS. Якщо необхідно, щоб при входженні в MPLS-мережу поле TTL не копіювалося з протоколу IP в полу TTL протоколу MPLS, використовується команда no mpls ip propagate-ttl. Тоді значення поля TTL MPLS встановлюється рівним 255, службова програма traceroute не вказує на проміжні вузли в MPLS-мережі, а відображає увесь домен MPLS як один IP-перехід.

2 Технології аналізу трафіку мережі інтернет-сервіс провайдера

2.1 розвиток технологій аналізу мережевого трафіку

Зародження технологій аналізу мережевого трафіку можна віднести до початку 90 х років минулого століття. Потреби в їх виникненні з'явилися приблизно в один час в декількох областях.

Ускладнення схем мереж і різноманіття мережевих пристроїв привели до ускладнення їх налаштування і підтримки мережі в працездатному стані - потрібен був інструмент який дозволяє, з одного боку локалізувати проблему, а з іншого надати якомога більш вичерпну інформацію про природу проблеми. Власне об'єктом, який містить в собі всю необхідну інформацію і є мережевий трафік. Одним з інструментів, спочатку призначеним для вирішення саме цієї проблеми став мережевий сніфер / аналізатор [3] Wireshark (раніше Ethereal), створений інженером Джеральдом Комбо (Gerald Comb) в 1997 році. Wireshark продовжує активно розвиватися і є стандартом в певній галузі мережевого аналізу.

В цей же час починає застосовуватися технологія трансляції адрес NAT , призначеної як для того, щоб заощадити IP адреси, так і для того, щоб приховати від зовнішнього спостерігача пристрій і ресурси внутрішньої локальної мережі. Для реалізації цієї технології був потрібний інструмент - апаратний або програмний транслятор адрес. Даний функціонал в результаті був впроваджений в якості складової частини в більшість маршрутизаторів. Існують і програмні реалізації, як в складі серверних операційних систем, так і у вигляді окремих додатків .

До цього ж часу відносяться перші згадки про віруси і DoS / DDoS атаки, в основному типу Syn flood - перша згадка про DDoS відноситься до 1996 року. Для захисту від цих загроз був потрібний інструмент, який аналізує і фільтрує пакети до їх потрапляння на основний сервер. Одним з видів таких захистів стали мережеві екрани (firewall). Перше покоління даних рішень відносилося до типу пакетних фільтрів (packet filters), які обробляли пакети по одному (не враховуючи передісторію) і аналізували тільки рівні L1-L3 моделі OSI і (для протоколів TCP / UDP) номера портів з транспортного рівня L4 (див. рис. 1). Для визначення типу

трафіку (web, email і т.д.) використовувався список фіксованих номера портів з каталогу IANA [4]. Процес аналізу полягав в порівнянні даних, витягнутих з пакета, з набором заданих правил і, в залежності від результату - блокування або пропуск пакету в мережу з занесенням події в журнал і опціональним повідомленням джерела пакета про ситуацію. Наприклад, правило «Блокування Telnet трафіку» виглядало, як правило, яке описує пакети, транспортний протокол яких - TCP, номер цільового порту - 23, а дія при виявленні такого пакета - блокування. Одним з перших подібних рішень був продукт DEC SEAL.

Ближче до кінця 90х - початку 2000х років, в зв'язку з ростом мережевих потоків даних, актуальними стали ще два завдання, які вимагали мережевого аналізу: балансування навантаження між серверами і прискорення роботи окремих видів мережевих додатків. До мережевих додатків, які вимагали прискорення, ставилися, перш за все, функції, які залежать протоколи HTTP, DNS, SSL . Для вирішення другої проблеми використовувалися, т.зв. проксі-сервера, які здійснюють кешування даних, що надходять, мінімізуючи, так чином, обміни по мережі.

Пристрої, розроблені для вирішення обох цих завдань носили назву контролери доставки додатків (Application delivery controllers, ADC). Такі рішення зокрема були розроблені компаніями Alteon, Radware, F5, Brocade, Cisco.

В першій половині 2000х років мережеві технології отримали бурхливий розвиток - з'явилися технології голосового обміну по мережі (VoIP) і обміну даними в тимчасових мережах P2P (Napster, KaZaA), що, зокрема, призвело до чергового різкого стрибка обсягів переданих по мережі даних. Для країн, що розвиваються, мереж великих корпорацій, потрібно об'єднувати в єдину локальну мережу територіально рознесені майданчики. Більш частими і складними стали мережеві атаки, що вимагало більш розвинених засобів захисту.

Для реалізації передачі керуючих сигналів і даних VoIP з використанням таких протоколів як SIP і RTP між різними провайдерами, як телефонного зв'язку, так і інтернету були потрібні спеціальні пристрої - прикордонні контролери сесій (session border controllers, SBC) , яким було потрібно виділяти відповідний трафік

із загального потоку. Дані пристрої проводилися в таких компаніях як Acme Packet, Audiocodes, Cisco, Genband.

Для вирішення проблеми ефективного обміну даними між різними сегментами розподіленої мережі, з'єднаними каналом обмеженої пропускної здатності (дана проблема має назву Channel optimization) був розроблений цілий спектр технік під загальною назвою Wan Optimizations . Серед цих технік можна вказати наступні

Дедуплікація (Deduplication) - зменшення повторної передачі даних за рахунок збереження на обох кінцях обміну повторюваних елементів даних і подальшої передачі посилань на ці дані замість самих даних. Може здійснюватися на різних рівнях мережевого стека (зокрема, TCP і IP).

Стиснення (Compression) - передача даних по каналу в стислому вигляді з подальшим розтиснену на іншій стороні.

Оптимізація латентності - упереджувальний відправка мережевих пакетів-підтверджень TCP.

Кешування одержуваного вмісту. Реалізовувалося за допомогою проксі-серверів, найбільш поширеними з яких були Web-проксі, кешуватися вміст сайтів. Прикладами такого ПО є Squid і NetCache.

Об'єднання декількох пакетів інтенсивних мережевих протоколів, таких як CIFS, в один (protocol spoofing).

Дані техніки згодом реалізовувалися як у вигляді окремих мережевих пристроїв (Middleboxes), так і програмно, на потужних серверах (Network appliances). Одним з перших виробників стала компанія Riverbed, згодом купила аналізатор Wireshark і інтегрувати його в свої продукти.

В сфері мережевої безпеки в цей період також відбулися значні зміни. Ускладнення мережевих атак призвело до того, що їх стало важко з достатньою точністю визначати по окремих пакетах, а швидкість появи нових атак - до необхідності реагування на ще невідомі їх види. У сукупності це призвело до появи методів захисту на основі аналізу поведінки мережевих потоків (tcp session behaviour analysis). У той же час стали з'являтися шкідливі сайти, що заражають їх

відвідувачів, а також методи впровадження шкідливого функціоналу в НЕ заражені сайти. Для захисту від таких атак було потрібно впровадження відновлювальних чорних списків сайтів і необхідність фільтрації і блокування по URL. Серед виробників засобів захисту можна вказати
Arbor, BlueCoat, SonicWall.

Найбільш повний розвиток технологія аналізу мережевого трафіку отримала, починаючи з другої половини 2000х років, в зв'язку з декількома факторами:
Безперервне зростання обсягів переданих даних.
Зростання ширини каналів, що забезпечують можливості для передачі цих обсягів.
Збільшення кількості різноманітності переданих даних, зокрема тих, які можуть використовуватися для складання різних профілів, як окремих користувачів, так і різних груп.
Зростання як різноманітності мережевих загроз і атак, так і їх кількісні характеристики.

Ці фактори призвели до зростання потреб з боку провайдерів інтернету (internet service providers, ISP) і різних компаній. Інтереси цих груп різні, але в той же час мають багато спільного.

Так, наприклад, загальної областю інтересів є захист мережевих ресурсів, яка, в свою чергу, ділиться на ряд напрямків:

— Антивірусні рішення (AV).

— Розвинені міжмережеві екрани Next Generation Firewalls (NGFW).

— Системи виявлення й запобігання мережевих атак Intrusion detection / prevention systems IDS / IPS.

— Системи захисту від DDoS-атак.

В водночас, специфічною областю інтересів провайдерів інтернету є [5]:

— Забезпечення якість зв'язку в години найбільшого навантаження (ГНН) з урахуванням економії на розширенні орендованих каналів зв'язку.

— Отримання конкурентної переваги за рахунок можливості пропонувати більш вигідні індивідуальні тарифи з урахуванням індивідуального профілю користування мережевим каналом.

— Регулювання смуги пропускання для деяких видів трафіку.

Однією з основних проблем є P2P трафік, який, може займати значну частину орендованого провайдером каналу (до 60-80% [6]), приводячи до того, що щоб забезпечити необхідну якість сервісу (quality of service, QoS) провайдеру доводиться прискореними (в порівнянні з прогнозами зростання абонентської бази и призначених для користувача потреб) темпами розширювати даний канал.

Основною областю інтересів компаній, що пропонують свої товари і послуги є використанням Інтернету, є «профілі» користувачів з точки зору їх інтересів і переваг. Подібні профілі можна опосередковано виявити, зокрема, за допомогою списку сайтів, які користувач відвідує, набору його пошукових запитів, мережевих додатків, які він використовує.

До іншої групи належать компанії, що надають різні інтернет сервіси, наприклад, за допомогою технології віртуалізації втратити зв'язок із мережею (Network Function Virtualization, NFV). До таких сервісів можна віднести: хмарні сервіси, сервіси захисту, зберігання і ін.

Для цих компаній, специфічним є питання управління великими обсягами вхідного трафіку - потрібно балансування та інтелектуальне управління.

Відповідно до наведеними вище історичним розвитком потреб в області мережевих сервісів відбувався розвиток технологій аналізу мережевого трафіку, які лягають в основу апаратних, програмних і гібридних рішень

2.2 Глибина аналізу мережевих пакетів

Технології аналізу трафіку розвивалися послідовно, кожна наступна успадковувала частина попередніх механізмів і додавала свої. Можна виділити три рівня розвитку технології, які наведені на рис. 1.

Розглянемо ці рівні більш детально.

2.2.1 Поверховий аналіз пакетів (SPI)

Технологія аналізу трафіку, яка грунтується виключно на заголовках пакету рівнів L1-L3 по моделі OSI. Пред'являє низькі вимоги до обчислювальних ресурсів, що дозволяє аналізувати великі обсяги трафіку.

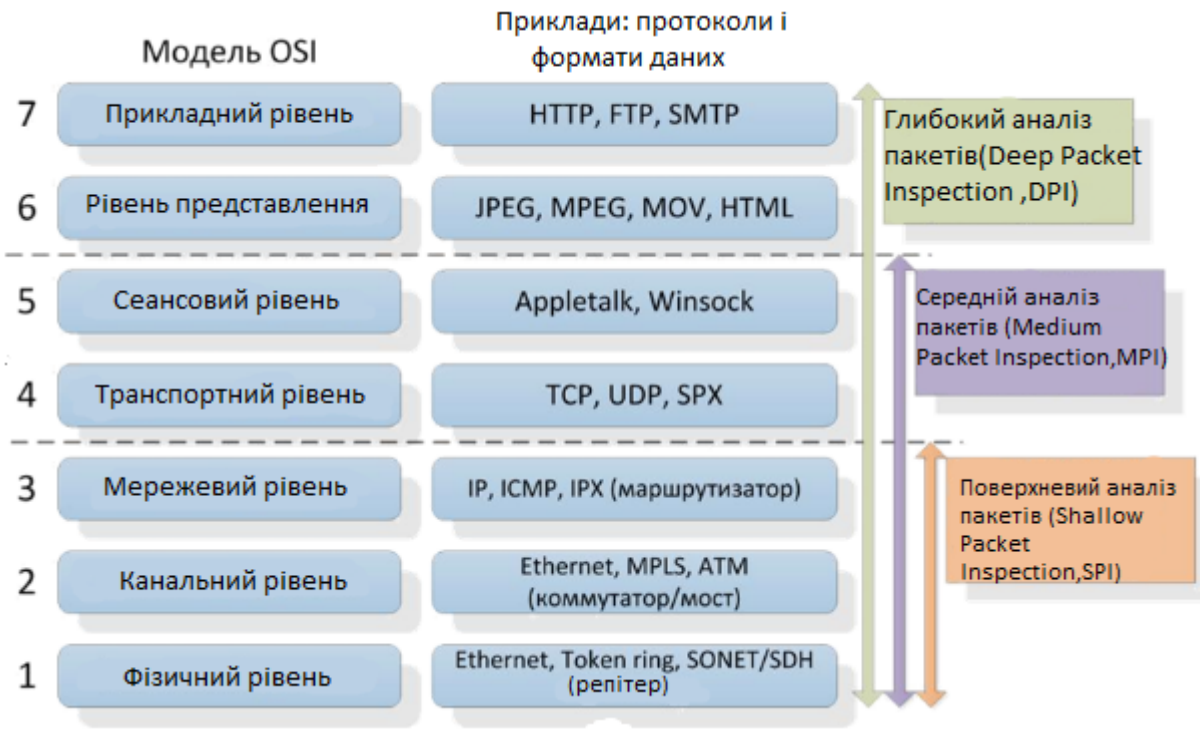

Рис. 2.1 - Рівні розвитку технології аналізу мережевого трафіку по «глибині».

Технологія широко поширена, на її основі працює більшість мережевих екранів операційних систем (зокрема в ОС Windows XP / Vista і OS X), маршрутизаторів і інших мережевих пристроїв. На її основі реалізовані мережеві списки контролю доступу на рівні IP адрес и портів (Access Control List, ACL). Таким чином, дана технологія добре підходить для розмежування доступу ззовні до окремих комп'ютерів (IP) і сервісів (порти) внутрішньої мережі.

2.2.2 Середній аналіз пакетів (MPI)

Технологія аналізу трафіку, яка грунтується на інспектуванні сесій і сеансів зв'язку, ініційованих додатком, але встановлюваних шлюзом-посередником (див. Рис. 2.2). Також застосовується термін «проксі додатків» (application proxy). В рамках даної технології вміст пакетів аналізується частково і по визначеним правилам. Не використовуються складні методи аналізу типу сигнатурного. Пристрої, що реалізують даний функціонал розміщуються між провайдером інтернету і кінцевим користувачем. Дані пристрої розбирають заголовки аж до транспортного рівня і невелику частину даних пакета для зіставлення розібраної

частини з деяким списком розбору (parse list), з подальшою реакцією в разі їх виявлення. Дані списки зазвичай коротше списків ACL і надають більш широкий діапазон дій на відміну від «дозволити / заборонити» в разі ACL. Ці списки також більш виразні, так як дозволяють прив'язуватися не до IP-адрес, а до формату даних пакетів і до даних деяких протоколів рівня додатків, наприклад, URL-адресами в разі протоколу HTTP. За допомогою MPI можна, наприклад, заблокувати можливість отримання flash-файлів або картинок з певних інтернет сервісів (на рівні представлення моделі OSI) або заблокувати частина команд (на прикладному рівні моделі OSI) в окремих протоколах. Набір протоколів, як правило, дуже обмежений. Наприклад, в перших версіях CheckPoint FireWall-1 (CheckPoint FW-1) підтримувалися протоколи Telnet, FTP, HTTP, а в Cisco Private Internet Exchange (Cisco PIX) - FTP, HTTP, H.323, RSH, SMTP і SQLNET. Згодом дані набори незначно розширювалися. Також відомо, що дана технологія використовується в продуктах компаній McAfee і Symantec. Міжмережеві екрани, що використовують цю технологію, відносяться до другого покоління .

Дана технологія більш гнучка в порівнянні з SPI і, крім розмежування доступу, підходить для більшого числа завдань - кешування вмісту, аналіз стисненого / шифрованого трафіку, обмеження функціонала окремих протоколів шляхом заборони окремих команд. Завдяки підключенню в режимі проксі, може служити в якості Wan Optimizer'a .

Основний недолік MPI - погана масштабованість: кожна команда і протокол вимагають окремого «шлюзу» (вхідний-вихідний порти). Крім того, робота в режимі проксі сильно знижує швидкість обробки. Для зниження навантаження на проксі-сервер був розроблений протокол ICAP , що дозволяє проксі-серверів відправляти проходять через них дані для проведення аналізу стороннім серверам на предмет безпеки або аналізу вмісту. Ця схема реалізована в антивірусному продукті ClamAV, який може підключатися до проксі-серверів Squid і NetCache, згаданим вище.

Ці фактори сильно обмежують застосування даної технології на рівні провайдерів інтернету внаслідок необхідності аналізу великого числа протоколів і команд на широких каналах зв'язку.

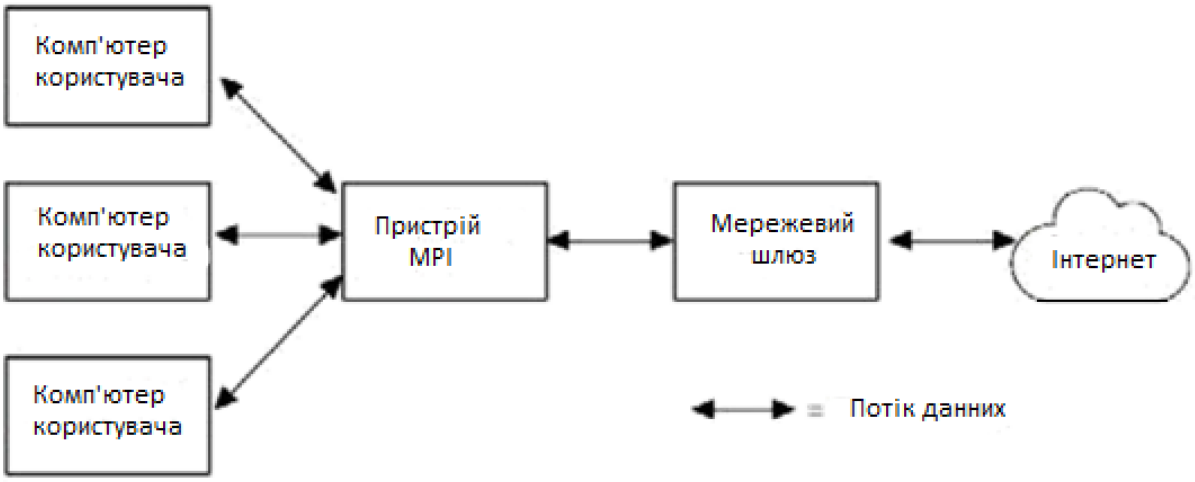

Рис. 2.2 - Схема застосування пристроїв аналізу на основі технології MPI.

2.2.3 Глибокий аналіз пакетів (DPI)

Іноді вживають більш вузький термін - DPP (Deep Packet Processing), який має на увазі такі дії над пакетами, як модифікація, фільтрація або перенаправлення. Сьогодні обидва терміни часто використовуються як взаємозамінні. Дана технологія є логічним розвитком MPI. В рамках даного підходу аналізатор переглядає вміст кожного пакета повністю. Одним з важливих відмінностей від попередніх технологій є те, що системи на базі DPI можуть приймати рішення не тільки по вмісту пакетів, але і за непрямими ознаками, властивим якимось певним мережевим програмами і протоколам. Для цього може використовуватися статистичний аналіз. Наприклад, аналіз частоти зустрічі певних символів, довжин пакетів, відстань між мітками часу послідовних пакетів і т.д. Також, в порівнянні з попередніми підходами.

На відміну від MPI, дана технологія спочатку розроблялася для високошвидкісної обробки та ідентифікації великої кількості додатків в реальному часі. Таким чином, рішення на основі DPI добре масштабуються як по ширині мережевого каналу (відомі рішення, що працюють на каналах близько 100 Гбіт /

сек), так і за кількістю ідентифікованих додатків (в існуючих рішеннях - порядку декількох тисяч). З точки зору реалізації, основний компонент будь-якого рішення DPI - модуль класифікації, що відповідає за класифікацію мережевих потоків. При цьому в залежності від цілей застосування DPI, класифікація може виконуватися з різною точністю:

— тип протоколу або додатка (наприклад, Web, P2P, VoIP)

— конкретний протокол рівня додатка (HTTP, BitTorrent, SIP)

— додаток, що використовує протокол (Google Chrome, uTorrent,Skype)

Важливо відзначити, що відповідність між класами різних рівнів точності неоднозначно, що показано на рис. 2.3.

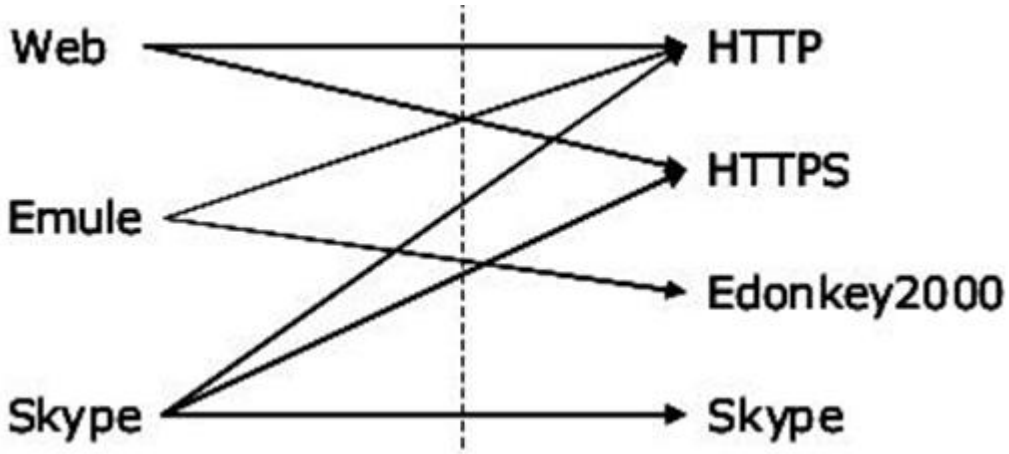

Рис. 2.3 - Різниця між ідентифікацією додатків (зліва) і протоколів (праворуч).

Технологія DPI на даний момент є поточним стандартом де-факто для засобів аналізу мережевого трафіку і відноситься до області критично важливих технологій необхідних для забезпечення, як мережевої безпеки, так і вимог законодавства. Внаслідок цього останнім часом на міжнародному рівні було прийнято низку стандартів, вимог і рекомендацій щодо особливостей реалізації, внутрішньою будовою та набору функцій відповідних можливостей . Ця технологія рідко застосовується в мережевих екранах - це скоріше область IDS / IPS систем, як винятків можна вказати екрани Hogwash і Shield. Однак міжмережеві екрани, які стосуються четвертого покоління можуть враховувати дані IDS / IPS систем в процесі аналізу..

2.3 Загальна схема інфраструктурних алгоритмів аналізу мережевого трафіку

Загальна схема аналізу мережевого трафіку складається з наступної послідовності кроків, кожен з яких призводить до підвищення рівня представлення об'єкта аналізу.

Захоплення пакетів, що проходять через контрольоване мережеве з'єднання. Результатом даного кроку є отримання об'єкта аналізу у вигляді мережевих пакетів. Залежно від необхідної точності і швидкості подальшого аналізу, а також доступних обчислювальних потужностей можуть використовуватися різні підходи.

Слайсінг (slicing), при якому аналізу піддаються не весь вміст пакетів, а тільки деякий префікс (n перших байт). У ряді досліджень показано, що цей підхід добре працює для подальшої класифікації трафіку по протоколах. В окремому випадку, якщо перехоплювати розмір дорівнює сумарному обсягу мережевих заголовків (L1-L3) є реалізацією технології SPI.

Самплинг (sampling), при якому перехоплюються не всі пакети, а тільки їх частина, яка може вибиратися за різними умовами, в залежності від потреб. В процесі розвитку технології було запропоновано велику кількість стратегій відбору. Наприклад, для завдань моніторингу типів трафіку підходить варіант з вибором кожного n-го пакета (uniform sampling), де n може вибиратися в залежності від співвідношення ширини каналу та пропускної здатності системи аналізу. Завдання отримання інформації про стані мережі за результатами самплинга відома як inversion problem , зокрема, при застосуванні uniform sampling відбувається недооцінка середнього розміру пакетів, так як частіше будуть відбиратися пакети меншого розміру. Для передачі перехоплених даних використовується протокол PSAMP.

Нарешті, для завдань, в яких потрібно максимально точний аналіз трафіку, наприклад для систем забезпечення мережевої безпеки, потрібно перехоплювати всі дані усього трафіку що надходить без втрат - для позначення цього підходу використовується термін lossless capture або deep packet capture (DPC).

Агрегування пакетів в потоки за деякими адресним ознаками (flow generaio), отримання нового об'єкта для аналізу - мережевого потоку. Якщо при цьому дані пакетів в подальшому аналізі не враховуються, то такий вид аналізу називається «аналіз потоків» - flow based analysis (на відміну від packet-based аналізу, при якому аналізуються дані пакетів). На рис. 2.3 показані відмінності типових схем packet і flow-based аналізу. Flow-based аналіз широко використовується в силу значно менших вимог до потужності обчислювача і пропускної здатності, за рахунок значного зниження обсягу даних для обробки. Такий вид аналізу може виконуватися як локально [7], так і віддалено від точки збору даних [8]. Для передачі зібраних даних від точки збору до точки аналізу використовується велика кількість протоколів, частина з яких стандартизована у вигляді IPFIX, а частина розроблена окремими виробниками - Cisco NetFlow, Juniper Jflow. В рамках підходу записи, що описують потік можуть містити різний набір даних. Найбільш загальним набором таких даних є наступний:

— IP адреси джерела і адресата,

— протокол транспортного рівня,

— в разі протоколів TCP / UDP - номери портів джерела / адресата,

— набір лічильників: кількість переданих пакетів і байт,

— час створення і завершення потоку.

Слід зазначити, що хоча даний метод дійсно значно знижує вимоги до аналізатору, тим не менш, він не є достатньо гнучким, так як на відміну від слайсинга і самплинга не дозволяє варіювати кількість даних, що надходять (воно залежить від вхідних даних). Більш того в більшості реальних задач кількість потоків незначно менше кількості пакетів (приблизно на порядок) через велику кількість дуже коротких потоків, що складаються з декількох пакетів - flash flows. Для вирішення цієї проблеми було запропоновано використовувати самплинг для потоків . Іншою особливістю даного методу є те, що, внаслідок обмеженості пам'яті, пристрій, що здійснює агрегацію пакетів, не може відстежувати один потік протягом довільного проміжку часу. Для вирішення цієї проблеми в конкретному рішенні зазвичай присутні налаштування, які обмежують максимальну тривалість

потоку (5 хвилин, в випадку Cisco NetFlow). Після закінчення цього часу вважається, що потік завершився, і інформація про подальші пакетах агрегируется в рамках «нового» потоку. Також в цій публікації описаний інструмент FLOW-REDUCE, який здійснює «збірку» повної інформації про потік з фрагментів, на які вона була розбита через обмеження за часом.

Виконання класифікації по протоколу прикладного рівня або конкретного мережевого додатком. Результатом даної операції є отримання нового об'єкта для аналізу - мережевого потоку конкретного протоколу або додатка (в цьому випадку пов'язаних потоків може бути кілька, наприклад, в разі VoIP додатки це потоки SIP і RTP). Після виконання даної операції можлива така додаткова обробка отриманого об'єкта, конкретний вид якої залежить від розв'язуваної прикладної задачі:

— розбір полів протоколу (protocol parsing),

— збірка сесії протоколу для протоколів з встановленням з'єднання,

— вилучення даних додатка (content extraction) - сторінок сайтів (HTML), файлів різних типів (виконувані, зображення, текстові документи, і т.д.), електронних листів, аудіо-відео потоків і т. д.,

— розбір даних програми (application content parsing).

Рис. 2.4 - Відмінності типових схем packet (зліва) і flow-based (праворуч) аналізу.

Для повноти картини, слід сказати, що крім зазначених вище packet-based і flow-based підходів існує ще одне джерело даних про трафік мережі - т.зв. база

керуючої інформації (Manage Information Base, MIB) - віртуальна база даних, яка використовується для управління об'єктами в мережі зв'язку.

Модулі для накопичення, зберігання та обміну даними в форматі MIB реалізовані в більшості пристроїв. Передача даних здійснюється по протоколу SNMP . Дані отримані таким шляхом мають низький обсяг і неспецифічні для протоколів. Наприклад, в рамках даного підходу, можна отримати відомості про загальну кількість пакетів і байт , які пройшли через конкретний мережевий інтерфейс конкретного мережевого пристрою.

Слід сказати, що однією з причин розвитку MIB і flow-based підходів, незважаючи на їх порівняно низьку точність, послужила досі актуальна глобальна дискусія про законність і допустимості глибокого аналізу трафіку з точки зору порушення безпеки, прав на приватне життя і т. д. На даний момент одним з наслідків даної дискусії є, зокрема, те, що в наукових роботах, трафік, який піддається глибокого аналізу попередньо проходить процедуру «анонімізації» за допомогою спеціальних засобів .

Далі будуть більш детально розглянуті окремі кроки з наведеної загальної схеми аналізу мережевого трафіку, методи, алгоритми та підходи, а також їх особливості та обмеження застосовності.

2.3.1 Захоплення мережевих пакетів

Програмні і апаратні засоби, які здійснюють захоплення трафіку відносяться до класу сніфера (sniffers). Для вирішення завдання захоплення трафіку можуть використовуватися як стандартні серверні мережеві карти, так і спеціалізовані мережеві карти, призначені для перехоплення трафіку на граничних швидкостях без втрат. Спеціалізовані карти, як правило, реалізовані на базі FPGA або ASIC і мають вбудовані засоби для проставлення тимчасових міток, апаратної фільтрації, зняття деяких заголовків низькорівневих протоколів, балансування навантаження між процесорами на багатопроцесорних комп'ютерах з урахуванням IP-потоків, виявлення помилкових і дублюються пакетів. При цьому вся обробка (в тому числі і копіювання даних в пам'ять комп'ютера з пам'яті мережевої карти) здійснюється

без залучення ресурсів ЦПУ. У міру розвитку технологій багато з описаних властивостей реалізуються і на базі стандартних мережевих карт. Технологія реалізації таких додаткових функцій носить назву TCP Offload Engine (TOE). Вона включає в себе різні технології, базовими з яких є наступні:

— Large Segment Offload (LSO) або Giant send offload (GSO);

— сегментація великих TCP-пакетів при відправці;

— Large Receive Offload (LRO) - збірка приходять окремих мережевих пакетів в великі сегменти;

— Checksum Offload - перевірка контрольних сум в заголовках;

— IPv4, IPv6, TCP і UDP;

— IP Security (IPSec) Offload - шифрування / дешифрування трафіку протоколу IPSec;

Основною проблемою для стандартних мережевих адаптерів є не швидкість передачі даних, як така, а кількість пакетів в одиницю часу. Це обумовлено особливостями внутрішньої реалізації обробників пакетів на мережевих картах, драйверів мережевих карт і програмних мережевих стеків ОС. Внаслідок цього, стандартні мережеві карти без спеціалізованих драйверів і мережевих стеків не забезпечують перехоплення трафіку без істотних втрат на швидкостях понад 3 Mpps (мільйонів пакетів в секунду). Причини такого обмеження будуть розглянуті нижче. Ще однією проблемою є точне проставлення тимчасових міток.

Проблеми, що виникають при переході до мережевим з'єднанням, що підтримує більш високі швидкості передачі даних, пов'язані в основному з декількома факторами:

— Обмеженою пропускною спроможністю апаратури.

— Архітектурними обмеженнями при взаємодії апаратури з ОС і ОС з користувацькими додатками.

— Об'ємом пам'яті, необхідним для зберігання одержуваних даних.

Більшість поширених систем аналізу трафіку працюють, використовуючи бібліотеки Libpcap (ОС Linux) і WinPcap (ОС Windows). Дані бібліотеки працюють в режимі користувача. Для забезпечення своєї роботи з боку ОС вони

використовують драйвери рівня ядра Berkeley Packet Filter (BPF) и Netgroup Packet Filter (NPF) відповідно. Основна різниця між цими драйверами полягає в схемі їх роботи з буферами пам'яті, що використовуються для тимчасового зберігання пакетів, одержуваних від мережевої карти. Драйвер BPF використовує схему з подвійною буферизацією, в той час як драйвер NPF використовує кільцевої буфер.

Серед проблем цих рішень, що призводять до зниження продуктивності можна виділити:

— Подвійне копіювання даних пакета (з карти в пам'ять ядра, з пам'яті ядра в пам'ять користувальницького процесу).

— Велике число переривань від мережевої карти (на кожен пакет, щоб він був скопійований в буфер ядра).

— Велике число перемикань між режимами ядра і користувача (на кожен пакет при його копіюванні в пам'ять користувальницького процесу).

— Недостатнє використання паралелізму на рівні окремих ядер і процесорів (за замовчуванням всі переривання обробляються одним ядром).

— Проблеми з синхронізацією при доступі до даних з декількох потоків виконання. У разі, якщо отримані дані повинні оброблятися в кілька потоків між цими потоками виникає ситуація змагання за ресурси.

В залежно від кількості копіювань даних пакетів, які виконуються в процесі перехоплення, рішення поділяються наступним чином.

— 0-copy (zero-copy). Для реалізації підходу з нульовим копіюванням потрібно апаратна підтримка з боку мережевої карти - вона повинна містити власний DMA контролер, що копіює дані з карти в пам'ять програми користувача, без додаткового копіювання через пам'ять ядра. Прикладом може служити бібліотека PF_RING ZC в зв'язці з мережевими картами Intel або Napatech

—1-copy. Для реалізації цього підходу можливі кілька варіантів - розробка аналізатора на рівні ядра, що є досить складним завданням або пряме відображення пам'яті ядра в пам'ять користувальницького процесу.

— 2-copy. Стандартне рішення на базі LibPcap або WinPcap.

Для вирішення перерахованих проблем було реалізовано кілька спеціалізованих драйверів і мережевих стеків, до яких відносяться, наприклад, комерційне рішення Sniffer10G від Emulex і Myricom, а також відкрита розробка PF_RING компанії Ntop. Ці рішення використовують схему з кільцевим буфером, як більш ефективну, а також оптимізовані для багатопроцесорних і багатоядерних комп'ютерів. Зокрема вони реалізують наступний функціонал:

— Обробка перехоплення пакетів з використанням великої кількості потоків виконання (один потік на вхідну чергу).

— Балансування навантаження між ядрами (одне ядро - одна вхідна чергу).

Для реалізації Пакетної фільтрації використовується як апаратна підтримка з боку архітектури, так і підтримка з боку ОС (спеціалізоване API). Серед використовуваних технологій можна виділити наступні.

Набір близьких технологій Interrupt Moderation, Adaptive Interrupt Moderation, Interrupt Coalescing, Interrupt Blanking, Interrupt Throttling, що дозволяють управляти затримкою доставки переривань за рахунок настроюваного таймера і обробляти отримання / відправку безлічі пакетів впродовж одного переривання.

MSI-X - розподіл I / O переривань по декільком процесорам і ядер.

New API (NAPI) - інтерфейс рівня ядра ОС Linux, що дозволяє застосовувати техніку зменшення кількості переривань (interrupt mitigation) з боку мережевих пристроїв.

Receive-side Scaling (RSS) - технологія, що надає можливість динамічного балансування навантаження входять мережевих пакетів по декільком ядрам і процесорам (переривання надходять на різні процесори). Існують реалізації для масштабування на випадки більше 64 процесорів. Дана технологія підтримується в сімействі ОС Windows з появою Scalable Networking Pack. В ОС Linux аналог цієї технології називається Linux Scalable I / O. Також існує ряд апаратних технологій від різних виробників процесорів, призначених для прискорення введення / виведення. Intel Integrated I / O - технологія прямого підключення шини PCI Express 3.0 до процесора (без окремого PCI-контролера), реалізована в сімействі Intel Xeon E5.

Direct Cache Access (DCA) - надання пристроїв введення / виводу, таким як мережеві адаптери, можливості переміщення даних безпосередньо в кеш процесора Intel.

2.3.2 Класифікація мережевого трафіку

Тема класифікації мережевого трафіку сама по собі є дуже великою. Перш ніж переходити до методів, якими вона здійснюється, перерахуємо варіанти класифікації за її результатами, тобто об'єктам, які виходять на виході даного алгоритму, їх властивостями і можливостям їх подальшої обробки. За цим критерієм, можна виділити три основні варіанти класифікації. Далі вони перераховані в порядку збільшення «точності» класифікації:

Тип трафіку не є достатньо змістовним способом класифікації і, як правило, або не береться подальшого аналізу, або піддається досить простий додаткової уточнюючої класифікації. Залежно від сфери застосування, типи можуть бути різними. Серед прикладів, можна вказати: P2P, відео-стрімінг, веб-трафік - в разі систем збору статистики і моніторингу, трафік мережевої атаки / нормальний трафік - в разі систем захисту від мережевих атак, трафік, що містить / який не містить об'єкти копірайту, в разі систем контролю копірайту.

Використовується протокол прикладного рівня (protocol identification) є досить змістовним і може, як використовуватися безпосередньо - наприклад, в системах збору статистики і моніторингу для підвищення рівня точності. Основним способом подальшої обробки є розбір протоколу, що включає два основних функції - складання сесії прикладного рівня, в разі необхідності вилучення даних протоколу з окремих його полів (метаінформація рівня протоколу).

Додаток, передає дані (application identification),дає максимально деталізований рівень класифікації. На цьому рівні можуть здійснюватися ті ж види обробки, що і на рівні протоколу прикладного рівня, а також вилучатись і інтерпретуватися дані (метаінформація) конкретного додатка, що відповідає більш високому рівню їх подання. Наприклад, поле типу «рядок», визначене на рівні протоколу, може відповідати «імені користувача» на рівні додатку.

В різних прикладних задачах результати ідентифікації протоколів і додатків можуть інтерпретуватися і, відповідно піддаватися різного подальшій обробці (як і в разі ідентифікації типу трафіку).

Наприклад, в разі системи захисту від шкідливого коду, під протоколом може розумітися командний (command-and-control, C & C) протокол ботнету, а під додатком - конкретний вірус. Відповідно, яку видобувають метаінформація - команди ботнету, передані їм дані, а мета аналізу - з'ясування його функціоналу, оцінка поширеності та дослідження можливостей його деактивації.

В випадку системи складання профілю користувача для подальшої демонстрації таргетированной реклами (наприклад, iMarker) в ролі протоколу може виступати HTTP, в ролі додатка - браузер, а об'єктом аналізу є запит користувача до пошукової системи, який піддається подальшому текстовому аналізу для вилучення ключових слів.

Вибір конкретної прикладної задачі може значно впливати як на вибір алгоритму класифікації, так і на його параметри і продуктивність. Як приклад можна розглянути наступне порівняння. У разі системи статистики, алгоритм класифікації зазвичай працює послідовно на пакетах кожного потоку «до першого спрацьовування». Схема такої класифікації наведена на рис. 2.5.

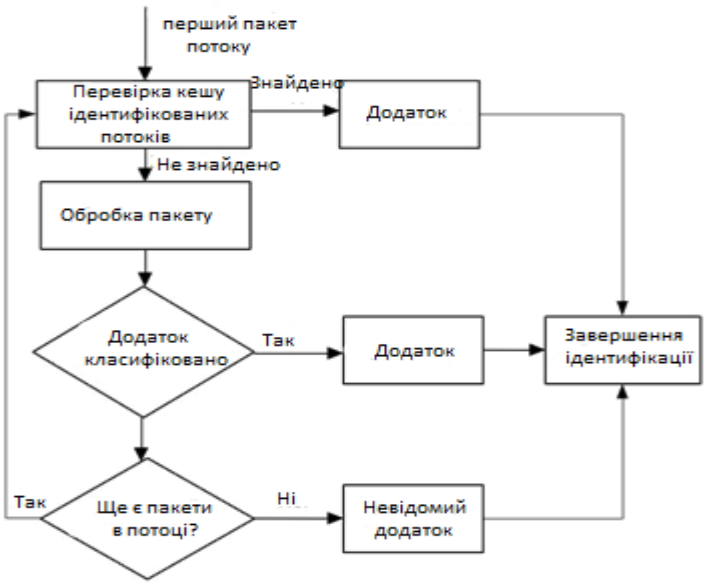

Рис. 2.5 - Схема класифікації «до першого спрацьовування».

В випадку систем фільтрації за ключовими словами такий метод не підходить, так як в одному і тому ж мережевому потоці, в різних пакетах можуть зустрітися різні слова і, з точки зору системи класифікації, в цьому випадку даний потік потрапить відразу в кілька класів.

В загальному випадку, очевидно, що перший підхід набагато продуктивніше, так як доводиться аналізувати значно менші обсяги даних. Крім того, в ряді підходів, для додаткового прискорення, аналізують не все вміст пакета, а тільки деякий його префікс (за аналогією зі слайсинга).

В роботі проведено оцінку впливу розміру аналізованого префікса пакету на точність класифікації по протоколам і швидкість роботи класифікатора на трьох знятих мережевих трасах Unibs-GT, Polito, Polito-GT.

На основі цих досліджень, зокрема робиться висновок про надмірності проведення IP-дефрагментації і TCP-нормалізації при вирішенні даного завдання, так як дані алгоритми (особливо другий) досить ресурсомісткі и практично не впливають на точність. Це відбувається через те, що для класифікації, як правило, використовується не більше 256 байт пакетів, а мінімальний розмір фрагмента зазвичай не менше 576 байт. Тобто, для даної задачі PBFS підхід більш кращий, ніж підхід MBFS .

Розглянувши види класифікації по отримуваних результатів і підходи в різних прикладних задачах, перейдемо до розгляду конкретних алгоритмів класифікації.

Класичним підходом до класифікації є аналіз вмісту пакетів (payload-based). При цьому, як правило, виконується пошук т.зв. «Сигнатур» (signature-based підходи) - характерних ознак, які заздалегідь створюються для кожної програми або їх груп. Класифікація може виконуватися як на рівні окремих пакетів (stateless аналіз), або може враховуватися стан потоку (statefull аналіз). Для підвищення точності розпізнавання частина підходів використовує уточнені «сигнатури» на основі автоматів станів протоколів. При такому підході, одержувані повідомлення, після їх класифікації, зіставляються з переходами в різних автоматах протоколів, і

оцінюється коректність послідовностей таких переходів. Ця група підходів називається Stateful Protocol Analysis Detection [6].

Класифікація є найбільш завантаженим алгоритмом аналізу мережевих пакетів. Історично, через нестачу обчислювальних потужностей, робилися спроби досягнення збільшення продуктивності алгоритму за рахунок вибору джерела даних, що використовуються алгоритмом в процесі класифікації, таким чином, щоб оброблювані дані, будучи не менш інформативними, ніж вміст пакетів, були б більш компактні. Ця група підходів (на відміну від «сигнатурного») відноситься до класу «заснованих на виведення» (inference-based).

Одним з важливих переваг inference-based підходів є те, що якість аналізу не залежить від представлення даних в мережевих пакетах, зокрема, відсутні обмеження при аналізі стисненого / шифрованого трафіку. Далі будуть розглянуті основні підходи до вирішення завдання класифікації, їх особливості та обмеження застосовності.

2.3.3 Класифікація мережевого трафіку на основі виведення

Всі підходи на основі висновку можна розділити на групи за двома основними параметрами:

- використовувані для виведення дані,
- використовуваний для їх аналізу алгоритм.

Всі види даних, в свою чергу, можна розділити на:

- характеристики окремих пакетів в рамках окремого потоку (Packet based),
- характеристики потоків в цілому (flow based).

До першої групи належать підходи, які використовують такі характеристики як: тимчасові проміжки між пакетами, послідовності розмірів пакетів [8], та ін.

До другої групи належать два основні підходи.

Підхід на основі аналізу портів (port-based) при якому ідентифікація відбувається по одному з номерів портів потоку, на основі бази даних про характерні статичних портах, які використовують зареєстровані в IANA протоколи (реєструвати можна будь-який номер порту, а не тільки перші 1024). Цей метод

вважається малоефективним, тому що на даний момент існує велика кількість протоколів з динамічними номерами портів. Зокрема, до таких протоколів відносяться практично всі реалізації P2P. Крім того, часто використовуються схеми, при яких трафік деякого протоколу (наприклад, HTTP) передається по невластивому для нього номером порту (не 80 в разі HTTP).

Підходи на основі статистичної інформація про активність окремих хостів в мережі: в скількох і яких саме обмінах даними (потоках) брав участь даний хост, скільки даних, і в який бік передавалося і т.д. Ці дані зіставлялися з набором заздалегідь створених шаблонів різних видів серверів. Один з таких підходів описаний в роботі.

Алгоритми аналізу даних діляться на два основних напрямки: порівняння з тим чи іншим видом заздалегідь створеного шаблону, підхід на основі машинного навчання і подальшого розпізнавання.

Методи на основі машинного навчання останнім часом отримали бурхливий розвитку. Однією з причин цього розвитку є доступність великої кількості різноманітних даних для навчання (соціальні мережі, великі БД, результати пошукових систем і т.д.). Ця група методів на даний момент представлена великим числом алгоритмів: байєсовські мережі, дерева прийняття рішень, методи опорних векторів, методи k-середніх і ін. Дані методи, в свою чергу діляться на групи за методом навчання, який застосовується для їх автоматичної інсталяції: класифікація (навчання з учителем), кластеризація (навчання без вчителя), асоціювання (association), чисельне пророкування (numeric prediction).

2.3.4 Класифікація мережевого трафіка на основі сигнатур

Недоліком цих методів є їх висока ресурсомісткість, пов'язана з необхідністю перегляду великих обсягів даних. Однак в даний час обчислювальні потужності дозволяють використовувати більш точні, ніж засновані на виведенні, сигнатурні методи, які, в свою чергу, діляться на дві великі групи:

— пошук рядків (string matching)

— пошук регулярних виразів (regexp matching).

— Сигнатури на основі рядків.

В процесі розвитку, для пошуку рядків застосовувалося велика кількість різних алгоритмів пошуку рядків, що володіють різними перевагами і недоліками, що визначало область їх застосування. Найбільш відомими алгоритмами є: прямий перебір (brute force, BF), Кнут-Моріс-Пратт (KMP), Бойєр-Мур (BM), Ахо-Корасік (AC), AC-BM (використовується в Snort), Wu-Manber, Commentz Walter (CW), фільтри Блума (Імовірнісна структура на основі хешу).

В роботі [9] проводиться огляд і порівняння великого числа методів пошуку рядків по тому як реалізований алгоритм порівняння з наявними сигнатурами. Виділено 4 групи методів:

— Послідовне порівняння з усіма сигнатурами (Exhaustive Search).

— Дерево порівнянь (Decision Tree).

— Декомпозиція (Decomposition), при якій окремі частини сигнатура обробляються незалежно, з наступним об'єднанням результатів.

— Асоціативний доступ (Tuple Space), при якому сигнатури розбиваються на групи біт, з якими проводяться операції порівняння.

На рис. 2.6 наведено розподіл значної частини алгоритмів за даними групам. Алгоритми, що лежать на кордонах, використовують гібридні підходи.

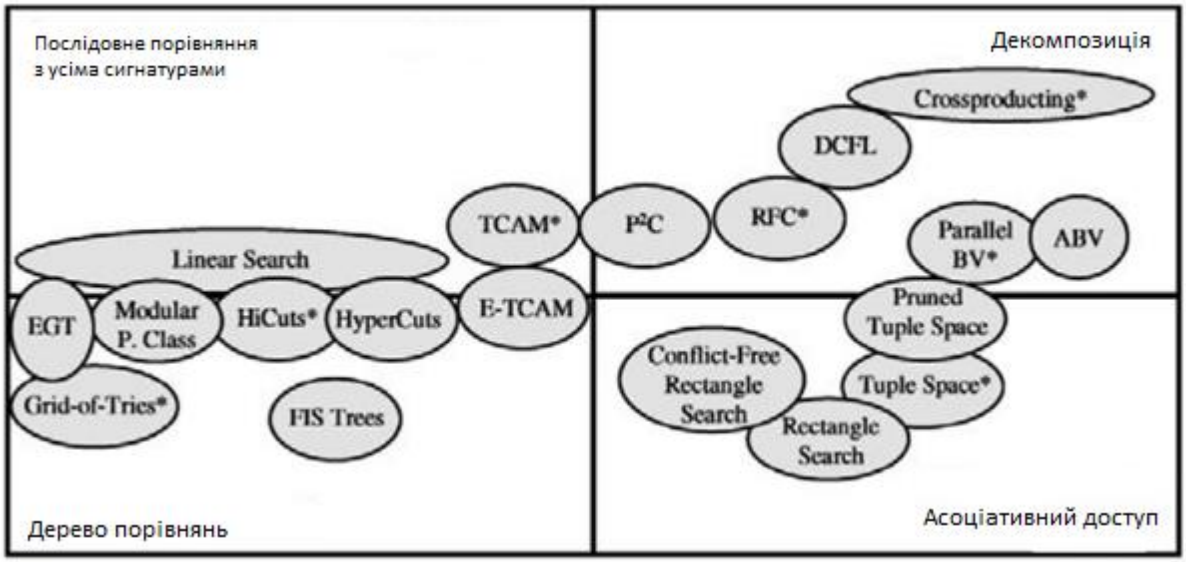

Рис. 2.6 - Розподіл алгоритмів пошуку строкових сигнатур за даними групам.

З ростом числа протоколів і їх складності строкове представлення було визнано недостатньо виразним, в зв'язку з чим, для опису сигнатур стали використовувати регулярні мови у вигляді граматик і регулярних виразів. Для ефективного пошуку сигнатур регулярний мову, що описує сигнатуру, представляються у формі кінцевого автомата. Виділяють два основних види автоматів - детерміновані або недетерміновані. Обидва ці уявлення мають свої переваги і недоліки.

Одна з відкритих баз сигнатур такого виду використовується у відкритій програмі для класифікації l7-filter . Крім того, такі підходи можуть не спрацьовувати в разі, якщо сигнатура була розділена на кілька пакетів на рівні IP або TCP. Для вирішення цієї проблеми, перед пошуком сигнатури необхідно виконати IP-дефрагментацію і TCP-нормалізацію відповідно.

Основною перевагою не детермінірованних кінцевих автоматів (НКА, NFA) є їх компактність: обсяг займаної пам'яті пропорційний числу символів, що входять в регулярні вирази. Однак для обробки кожного символу вхідних даних недетермінованим кінцевим автоматам може знадобитися до O (N) звернень до пам'яті, де N - число станів автомата . З цієї причини можливості застосування НКА в високонавантажених системах обмежені.

В свою чергу, детерміновані кінцеві автомати (ДКА, DFA) вимагають для кожного вхідного символу зробити єдине звернення до пам'яті. Їх використання може представляти труднощі в зв'язку з їх великим розміром: число станів ДКА може експоненціально зростати («експонентний вибух»), і обмежена O (2l), Де l - сумарна довжина регулярних виразів в канонічному поданні. В роботі [10] було проведено дослідження впливу різних типів регулярних виразів на зростання розмірів автомата. Результати показані на рис. 2.7. Було виділено 3 типи регулярних виразів, з точки зору їх впливу на розмір автомата:

— вираження, прив'язані до початку пакета (пошук здійснюється тільки на початку пакета);

— вираження, прив'язані до початку пакета і містять зірочку Кліні ;

— вирази, які не прив'язані до початку і містять зірочку Кліні .

— виразів із зірочкою Кліні.

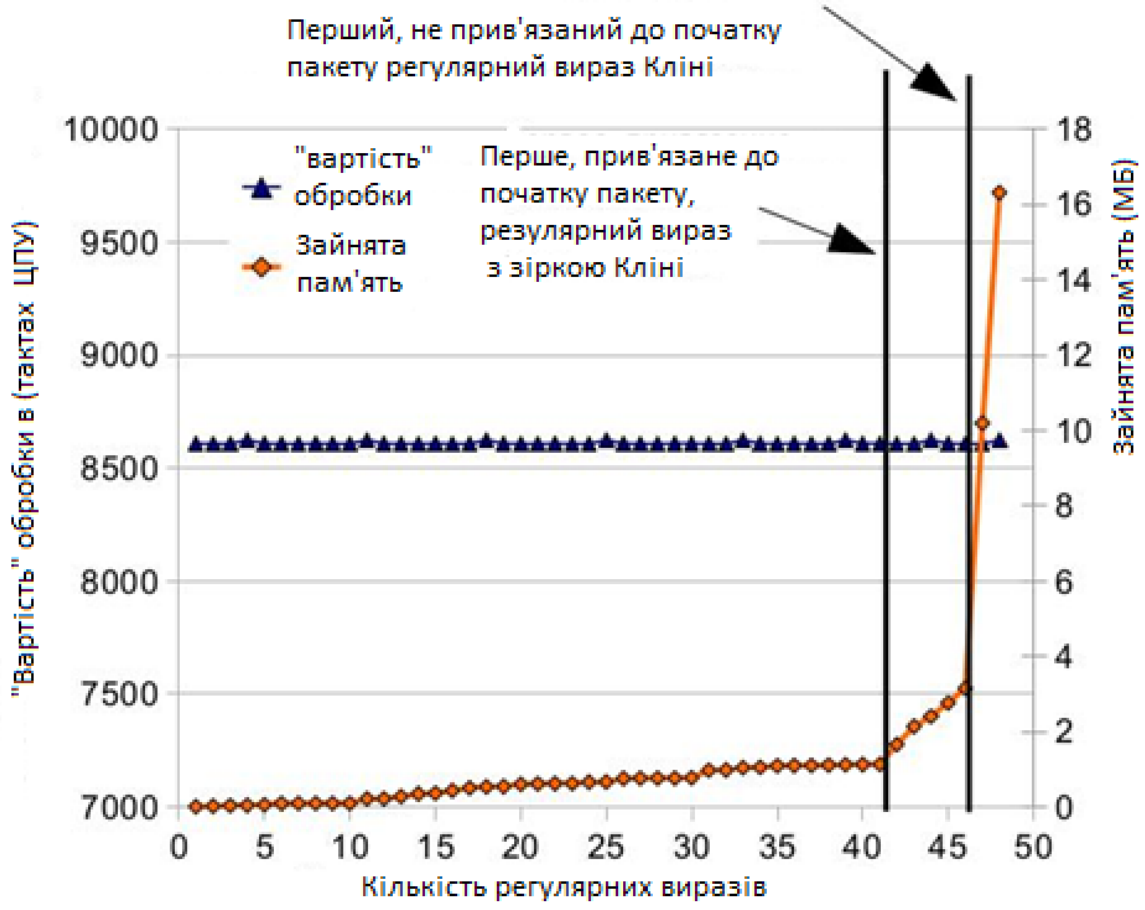

Рис. 2.7 - Експоненціальне вибух розміру DFA при додаванні регулярних

Для зниження розмірів автоматів часто застосовують різні види стиснення. Такі автомати звуться стислі ДКА (Compressed DFA, cDFA). У табл. 1 наведено порівняння трьох основних видів автоматів за розміром і продуктивності пошуку, взяте з роботи. Дані автомати були побудовані за регулярними виразами класифікатора L7. Для запобігання експоненціального зростання розміру, детерміновані автомати були розділені на 4 частини.

Незважаючи на проблеми з вимогами до пам'яті, детерміновані кінцеві автомати (і їх модифікації) отримали набагато більшого поширення в високошвидкісних системах аналізу. Незважаючи на проблеми з вимогами до пам'яті, детерміновані кінцеві автомати (і їх модифікації) отримали набагато більшого поширення в високошвидкісних системах аналізу.

Таблиця 1.1 - Порівняння розмірів і швидкості роботи основних видів кінцевих автоматів.

алгоритм	Вартість в тактах ЦПУ (Хв, пор., Макс.)	кількість автоматів	розмір автоматів
НКА	$2.2 * 10^4$, $4.1 * 10^7$, $8.9 * 10^7$	1	509 КБ
ДКА	52, $2.5 * 10^4$, $3.6 * 10^4$	4	230 Мб
стиснутий ДКА	268, $1.2 * 10^5$, $1.7 * 10^5$	4	53 Мб

Сучасні системи аналізу трафіку висувають високі вимоги, як до швидкості обробки даних, так і до кількості регулярних виразів, задіяних в обробці і, відповідно, розміру підсумкового автомата. Так як ні ДКА, ні НКА не можуть задовольнити одночасно вимоги і за швидкістю, і за розміром пам'яті, в даний час ведеться велика кількість досліджень по розробці гібридних уявлень. З точки зору реалізації, автомати являють собою таблиці з станів, в кожному осередку яких знаходиться список можливих переходів з цього стану в інше. Тому два основних напрямки робіт зосереджені на зменшенні числа станів і переходів відповідно.

2.3.5 Аналіз даних в різних додатків

Одну з важливих проблем для класифікаторів на основі вмісту представляє той факт, що одні й ті ж дані (наприклад, рядок) можуть бути при передачі по мережі бути закодовані по-різному, в залежності, наприклад, від використовуваного протоколу. Зокрема, під «різними додатками» в даному розділі є такі аспекти.

Різні методи кодування, зокрема для текстових даних - ASCII і Unicode кодування, а для бінарних даних - різні транспортні кодування, наприклад уявлення у вигляді тексту (Binary-to-text), прикладом яких є Base64.

Стиснення даних для зменшення завантаженості каналів передачі даних, наприклад використання gzip і deflate алгоритмів для стиснення вмісту HTTP-повідомленні.

Шифрування даних для забезпечення безпеки, наприклад використання криптографічних алгоритмів RC4 і AES в протоколах SSL / TLS.

За даними різних досліджень, стислий і зашифрований зв'язок (іноді використовується загальний термін «непрозорий», opaque) становить все більшу частку від усіх мережевих потоків даних . Це є наслідком великої кількості факторів, таких як:

— зростання популярності онлайн відеосервісів, що використовують стиснення відеопотоків,

— поширеність P2P-сервісів, які в більшості своїй використовують шифрування,

— використання шифрованого з'єднання (HTTPS) за замовчуванням на багатьох популярних сайтах,

— впровадження стиснення в HTTP протоколі на багатьох Web-серверах.

Проблема класифікації цих видів трафіку має кілька аспектів.

Для коректної класифікації такого трафіку потрібно додатковий функціонал. Спроба класифікувати такий трафік «в лоб» істотно знижує загальну продуктивність класифікатора, так як доводиться переглядати всі дані пакетів, проходячи по великій частині автомата і при цьому результат майже напевно буде негативним. Тобто такий трафік є «найгірший випадок», характеристики роботи на якому алгоритмів класифікації істотно гіршими за середні .

Для вирішення першого аспекту проблеми використовуються кілька підходів:

Генерація копій сигнатур, які піддаються різним видам стиснення і кодування. Даний метод обмежений тільки деякими алгоритмами стиснення і кодування, а також погано масштабується з урахуванням зростання кількості алгоритмів стиснення і їх кількості їх параметрів.

Використання модулів, які здійснюють разжатие / перекодування даних перед їх класифікацією. Цей метод має такі ж обмеження, як і попередній і також погано масштабується. Крім того цей метод збільшує вразливість системи до атак типу zip bomb, при яких розмір розтискати даних перевершує розмір стислих на кілька порядків.

Установка системи аналізу на місці або після засобу, який здійснює разжатие / розшифрування даних. Приклад такого засобу - проксі-сервер.

Для усунення другого аспекту, потрібно подавати на модуль класифікації трафіку тільки «прозорий» трафік, для чого з усього трафіку потрібно попередньо відфільтрувати «непрозору» його частина. Для вирішення цього завдання розроблені алгоритми, велика частина яких використовує характерну властивість «непрозорого» трафіку - підвищену ентропію значень його окремих байт. Приклади таких алгоритмів наведені в роботах [11].

3 Балансування трафіку з використанням технології MPLS на основі глибокого аналізу трафіку

3.1 Загальна концепція реалізації

У сучасних умовах інтернет-сервіс провайдеру для конкурентного існування на ринку замало надавати тільки послугу доступу до всесвітньої павутини. Також провайдер в своїй мережі надає доступ до різноманітних сервісів, інтерактивного телебачення, або ж може зі своєї сторони захищати абонентське підключення від різноманітних загроз з зовні.

Для реалізації будь-якого додатково функціоналу потрібне впровадження додаткового обладнання, яку підтримує спеціалізовані технології та протоколи роботи з різноманітними службами.

Технологія DPI надає широкий спектр послуг, які можна реалізувати за її допомогою. Від надання спеціалізованих платних послуг і обліку трафіку використаного окремо кожним додатком , до забезпечення найвищої якості обслуговування в мережі з комутацією пакетів.

Технологія DPI дає багато корисних можливостей , які детально були описані в другому розділі. Але великим недоліком цієї технології є те що при глибокому аналізі даних збільшується затримка на обробку , як наслідок при проходженні усього шляху від відправника до отримувача затримка може бути критичною. Для того щоб вирішити цю проблему було вирішено разом з технологією DPI використати багато-протокольну комутацію по міткам.

В загальному до структури мережі є наступні вимоги:

— на кожному маршрутизаторі , в який входять зовнішні канали потрібно встановлювати систему DPI

— на кожному маршрутизаторі, який є шлюзовим для певного сегменту мережі потрібно встановлювати систему DPI

— кожний маршрутизатор має підтримувати технологію MPLS

Дана реалізація актуальна тільки для мереж з великою

надлишковістю, яка забезпечує високу відмовостійкість, але як наслідок збільшує час проходження даних по мережі.

Головне і найскладніше завдання , це налаштувати маршрутизатори Core рівня. На них має бути налаштовані всі додаткові послуги і функції. Також маршрутизатори цього рівня представляють собою обладнання , яке підтримує технологію MPLS. Отже, проаналізувавши трафік згідно певних правил ми передаємо управління над маршрутизацією протоколу MPLS, який в свою чергу відправляє дані згідно своєї таблиці маршрутизації.

На маршрутизаторах доступу реалізований подібрий функціонал.

Також можлива схема включення без DPI на маршрутизаторах доступу, але тоді QoS буде реалізовуватися тільки силами MPLS, що дасть певний виграш в швидкості обробки пакетів, але не такий суттєвий.

3.2 Практична реалізація

Ми маємо складну топологію мережі великого інтернет-сервіс провайдера(рис 3.2), який забезпечує безперебійний доступ до послуги, як наслідок має надлишковість в мережі

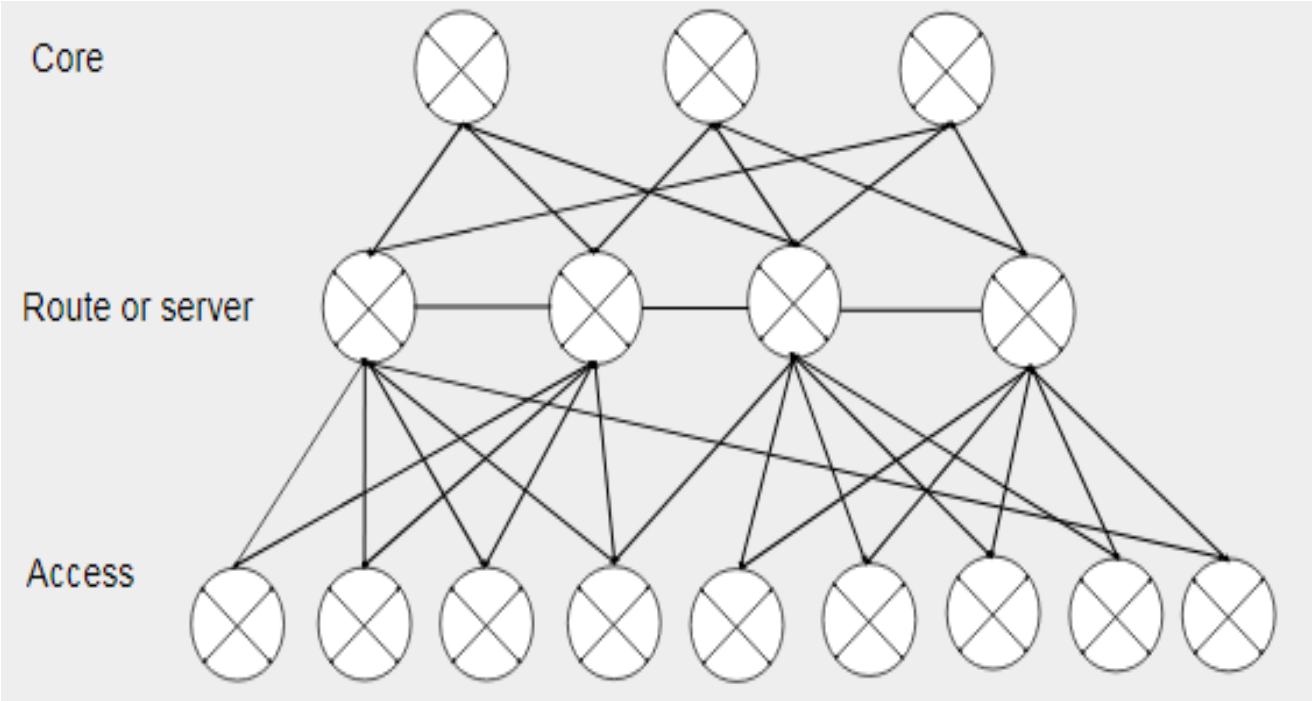

Рис 3.1 - Стартова топологія мережі

На основі цієї мережі потрібно реалізувати наступні послуги:

— встановити кеш-сервер компанії Google,

— встановити кеш-сервер компанії Facebook,

— налаштувати QoS,

— налаштувати захист від DDoS атак на абонентів і обладнання провайдера,

— налаштувати облік трафіку який був скачаний абонентом при використанні власного потокового сервісу.

Перший етап це встановлення додаткових серверів, для збереження(кешування) та ще один для потокового сервісу.

Розглянемо наближено роботу кеш-серверу. Звернувшись до певного ресурсу, наприклад youtube.com абонент отримає IP-адрес нашого кеш-серверу. Тепер всі запити на цей сайт будуть оброблятися нашим сервером. Коли якийсь абонент відправляє запит на перегляд певної сторінки, цей запит приходить на наш сервер і перевіряється чи є на нашому сервері потрібні дані. Якщо хтось в нашій мережі нещодавно робив запит до цих даних, тоді вони будуть збережені на сервері. Якщо ж потрібних даних на сервері немає, тоді запит перенаправляється на зовнішній сервер компанії Google , на якому точно є потрібні нам дані і вони завантажуються на наш кеш-сервер, який дозволяє завантажити потрібні дані абоненту.

При вході нашого кеш-сервера з ладу , зразу ж перебудуються маршрути і абонент буде отримувати IP-адресу серверів компанії Google. В момент переключеня можливе пропадання доступу до ресурсу до десяти секунд.

Кеш-сервер має бути в одному екземплярі для однієї послуги, оскільки , якщо він вийде з ладу абоненти цього не відчують. Хоча це і призведе до додаткового навантаження на мережу і зовнішні канали.

На відміну від кеш-сервера , для забезпечення потокового сервісу потрібно встановити декілька серверів в різних дата-центрах, що дозволить балансувати навантаження і в разі проблем з одним з серверів, інші можуть виконувати його роботу. Для достанньої надійності було вирішено встановити три таких сервера.

Після встановлення нових серверів топологія мережі буде мати наступний вигляд, див рис 3.2

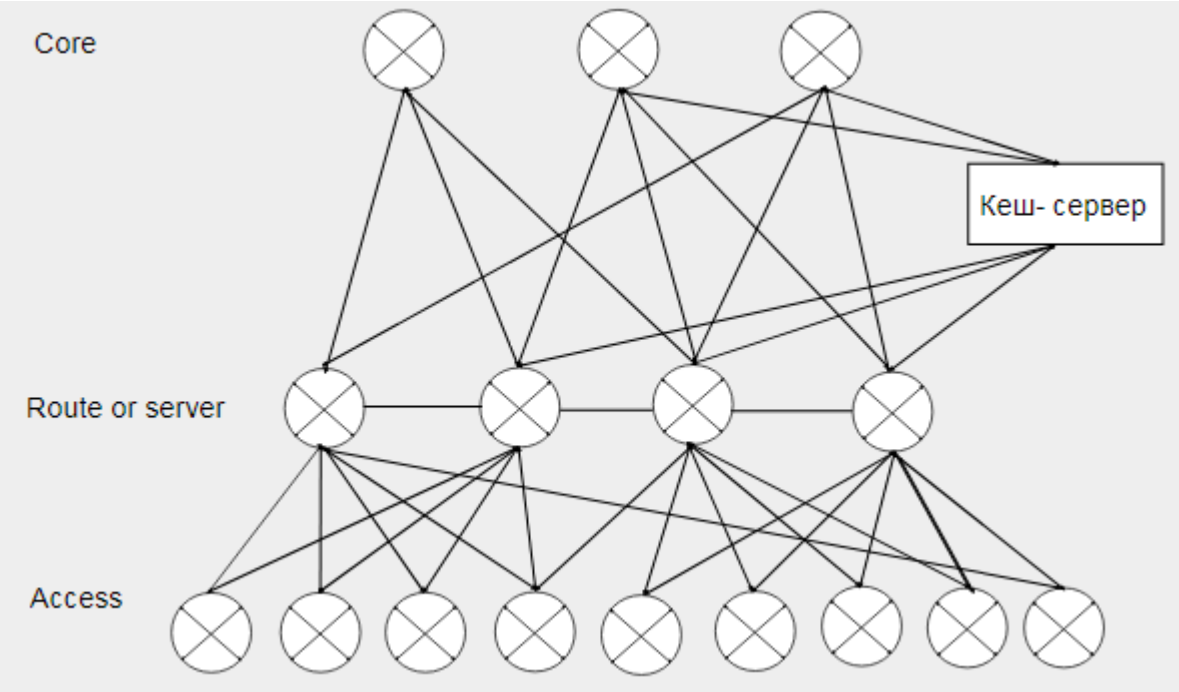

рисунок 3.2 - Топологія мережі з додатковим кеш-сервером

Далі нам потрібно провести налаштування на маршрутизаторах рівня доступу. Потрібно проаналізувати трафік, який приходить від абонентів і згідно власних правил якості обслуговування інкапсулювати кожний пакет міткою протоколу MPLS. Аналіз трафіку проводимо за допомогою бібліотеки OpenDPI, де налаштовуємо наступні правила :

Наступним етапом потрібно інкапсулювати кожну групу трафіку різними мітками MPLS і кожній групі надати власний пріоритет проходження по мережі, як наведено в таблиці 3.1.

Таблиця 3.1 - Мітки MPLS для різних груп трафіку

Група трафіку	Мітка MPLS	Пріоритет
AIM	12	43
Google Talk	14	48
Netgeat EVA	15	46
SSH	84	68
Telnet	64	63
VPN	23	61
FTP	87	43
SMTP	41	25
WWW	28	42
DNS	94	49
ICPM	44	19
eMule / eDonkey	31	45
Kazaa	64	46
Gnutella	79	45
BT / Azureus	84	41
Torrent	51	21

Діапазони пріоритетів трафіку поділені на класи для більшої зручності при налаштуванні, як наведено в таблиці 3.2

Таблиця 3.2 - Діапазон пріоритетів

Клас трафіку	Діапазон груп	
Клас A	0-9	Нехарактеризований трафік
Клас B	10-19	Заповнюючий трафік(приклад, мережеві новини)
Клас C	20-29	Несуттєвий інформаційний трафік
Клас D	30-39	Резерв
Клас E	40-49	Суттєвий трафік (FTP, HTTP, NFS)
Клас F	50-59	Резерв
Клас G	60-69	Інтерактивний трафік (telnet, SSH)
Клас H	70-79	Керуючий трафік (протоколи маршрутизації , SNMP)

Пакети даних з відповідними мітками передаємо протоколу MPLS для подальшої маршрутизації по мережі.

На маршрутизаторах рівня CORE проводимо аналогічні налаштування фільтрування трафіку по групах з розподіленням пріоритетів для груп як на маршрутизаторах рівня доступу. Також потрібно захист нашої мережі від DDos-атак.

Для здійснення DDos атак зловмисник має з розпорядженні велику мережу віддалено керованих комп’ютерів (BOTNET) і йому вже не потрібно приховувати IP-адресу кожного з них. В такому випадку зловмисник може просто імітувати дії справжніх користувачів сайту. Але завдяки великій кількості комп’ютерів, які беруть участь в атаці, навіть такі дії можуть викликати велике навантаження на сервер і ризвести до відказу в роботі. Зазвичай зловмисники вибирають для виклику найбільш ресурсомісткі запити, щоб мінімізувати чисто учасників в атаці комп’ютерів, IP-адреси яких після атаки будуть засвічені.

Часто для захисту від подібних атак з різною мірою ефективності приміняють різноманітні поведінкові стратегії (behavioralDDoSprotection), які дозволяють

визначити відхилення в нормальній поведінці. В нас реалізований простіший метод - використання тесту Тюрінга, комп'ютерного тесту, використаного для того , щоб визначити ким є користувач системи: людина чи комп'ютер.

Захист працює наступним чином:

— при перевищені порогового значення , наприклад комфортного для сайту кількості запитів в секунду,активується захист;

— до роботи з сайтом допускаються тільки користувачі , які знаходяться в білому списку, решта перенаправляються на сторінку CAPTCHA для перевірки на "людяність". Ця сторінка розташована на окремому сервері і здатна витримати нагрузку BOTNET будь-якого розміру;

— користувачі, які успішно пройшли тест добавляються в білий список і можуть працювати з потрібним ресурсом;

— користувачі які не пройшли тест не можуть пройти далі сторінки з CAPTCHA і не можуть створювати навантаження на ресурс.

Атака SYN flood викликає підвищені витрати ресурсів , так як на кожний вхідний SYN пакет система повинна зарезервувати певний ресурс в пам'яті , або згенерувати спеціальну SYN+ACK відповідь, яка яка включає криптографічний cookie, здійснювати пошук в таблицях сесій і так далі. Тобто витратити суттєві процесорні ресурси. В обох випадках відмова в обслуговуванні настає при потоці SYN-flood 100000-500000 пакетів в секунду. При чому навіть гігабітний канал дозволить зловмиснику направити на атакуючий сайт потік до 1,5 мільйонів пакетів в секунду.

Послідовність дій при захисті від SYN flood:

— виявити атаку по перевищенню заданого порогу непідтверджених клієнтом SYN запитів;

— самостійно , замість ресурсу, на який направлена атака відповісти на SYN запити;

— організувати TCP сесію з ресурсом, який ми захищаємо після підтвердження запиту клієнтом.

3.3 Результати впровадження

Загалом ми маємо можливість збирати найрізноманітнішу статистику в нашій мережі. Найбільш інформативними є наступні дані:

— загальний трафік розділений по типах, див рисунок 3.3., де показано розбиття всього трафіку по типах, з якого більшу частину складає веб-трафік, потоковий трафік та з'єднання рівний до рівного(peer-to-peer)

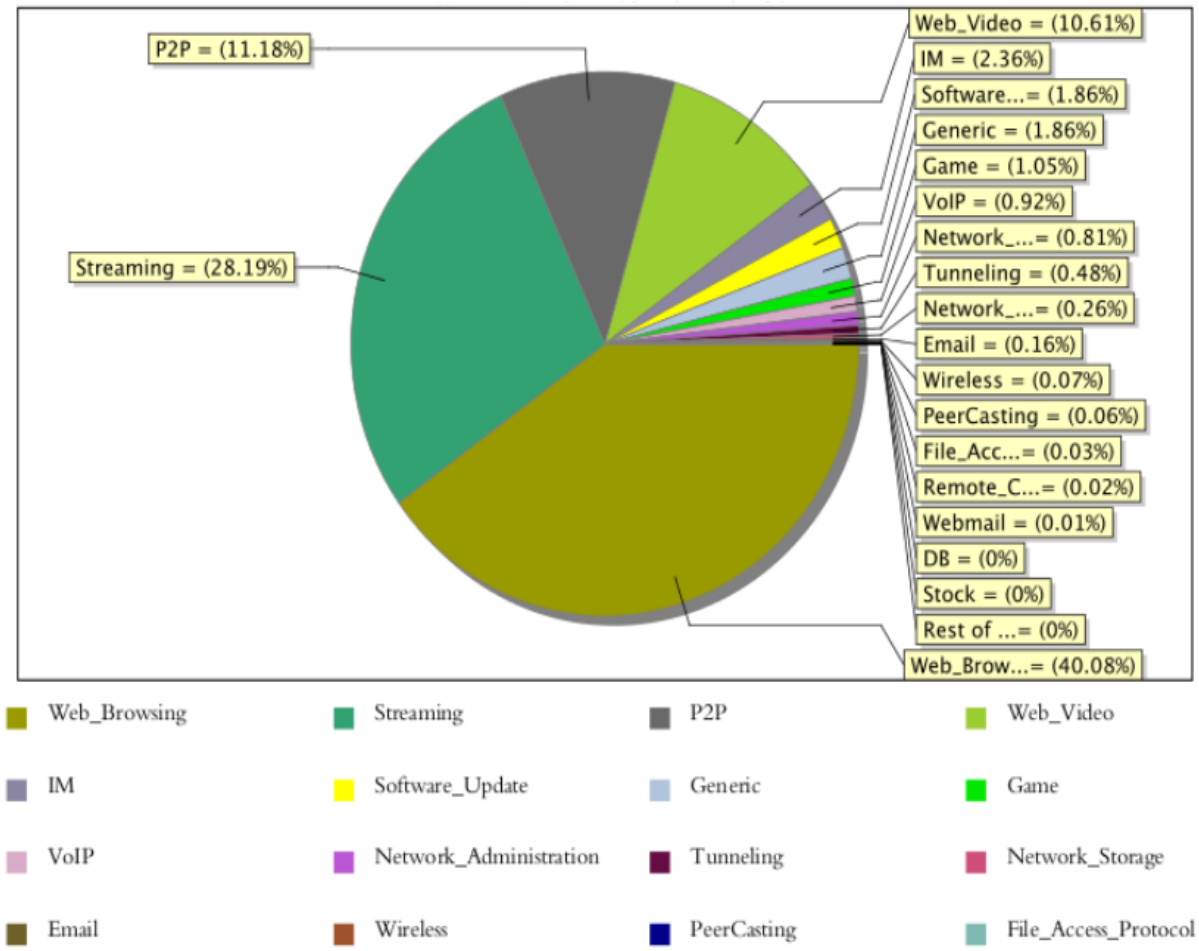

Рисунок 3.3 - Загальний трафік розділених по типах

— які відео хостинги найбільш популярні, див рис 3.4, де показано, що найбільш популярний відеохостинг - це сайт youtube.com.

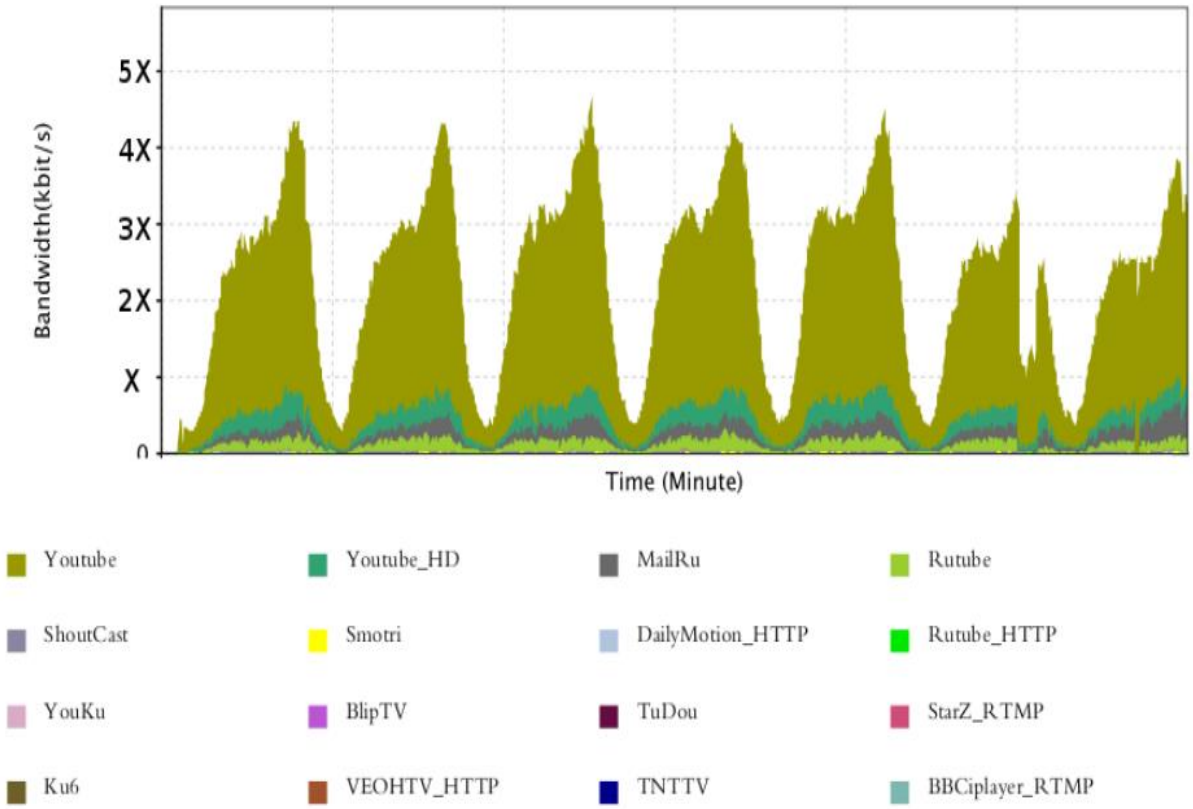

Рисунок 3.4 - Найбільш популярні відеохостинги.

- сайти, які найбільше споживають трафіку, див на рисунку 3.5, де безаперечне лідерство тримає сайт youtube.com. оскільки він дуже популярний і з нього завантажують в більшості відео, яке займає великі обсяги пам’яті

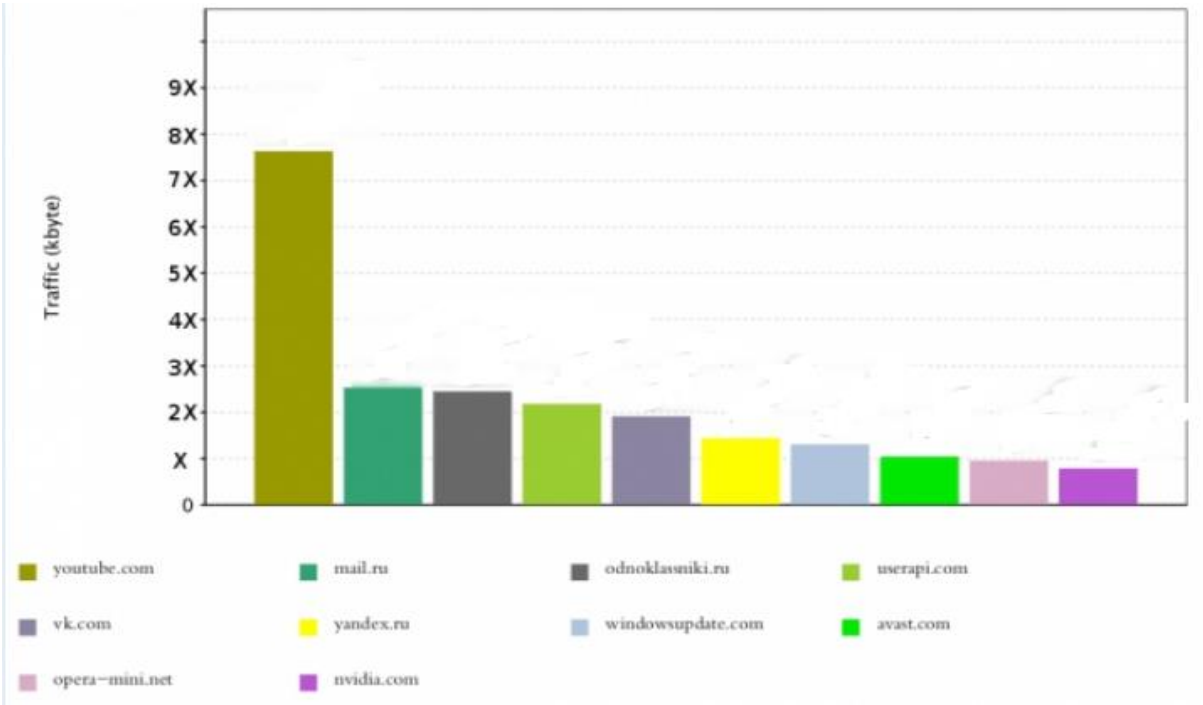

Рисунок 3.5 - Топ 10 сайтів по об’єму трафіку.

Встановлення додаткового кеш серверу не просте і дороговартісне завдання, але результати того варті. Після встановлення кеш-серверів в нашій мережі ми спостерігаємо наступну картину:

- трафік, який на себе перетягнув кеш-сервер компані Google , див рисунок 3.6

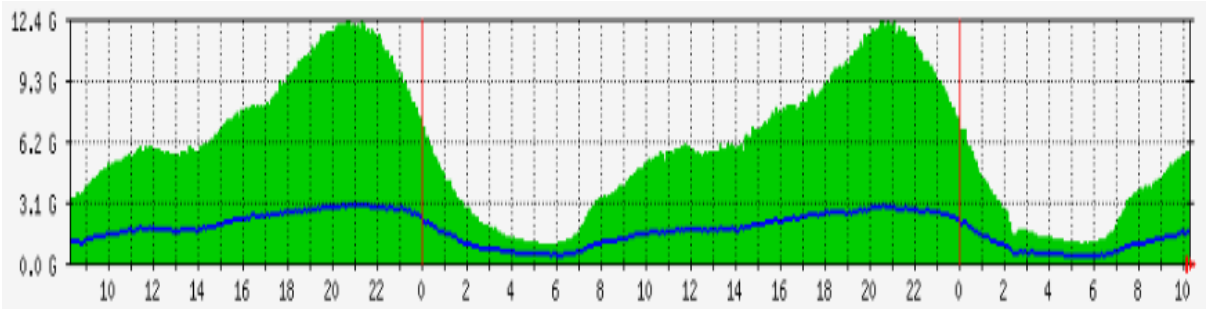

Рисунок 3.6 - трафік, який обробляє кеш-сервер.

- навантаження на решту зовнішніх каналів, див рисунок 3.7

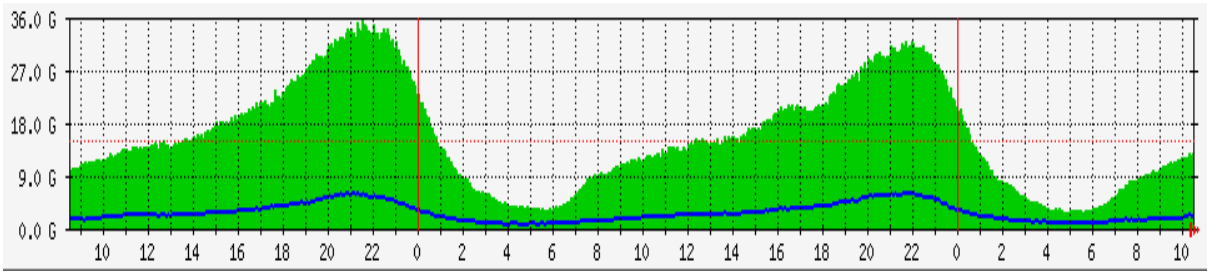

Рисунок 3.7 - навантаження на зовнішні канали.

В результаті ми отримуємо економію ємності зовнішніх каналів на 12.4 Гігабайт в секунду, або ж 25.8% за формулою (3.1)

$$P = \frac{N}{N+M} \times 100\% = \frac{12.4}{36+12.4} \times 100\%, \qquad (3.1)$$

де P - відсоток економії ємності зовнішніх каналів;

N - завантаження кеш-сервера;

M - загальне завантаження зовнішніх каналів (після встановлення кеш-сервера).

Після налаштування QoS , навантаження в пікові години на мережу також знизилися і ми отримали наступні результати, див рисунок 3.8

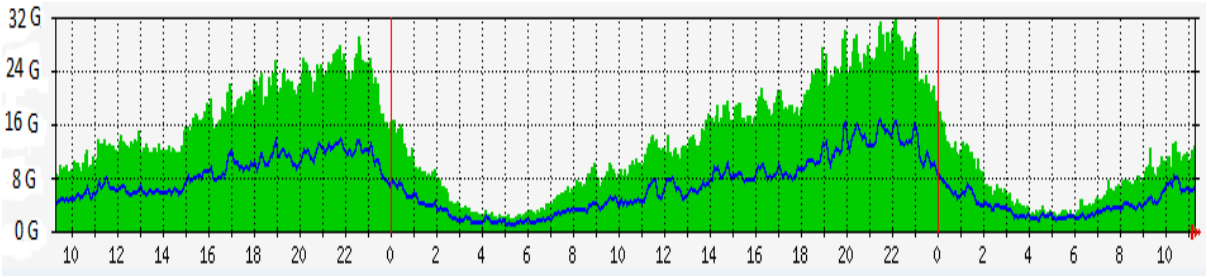

Рисунок 3.8 - результати впровадження QoS

Загалом за формулою (3.2) ми отримали виграш в 33.88%, або ж на 16.4 Гігабіт в секунду.

$$P = \frac{M-L+N}{N+M} \times 100\% = \frac{36-32+12.4}{36+12.4} \times 100\%, \qquad (3.2)$$

де L - завантаження зовнішніх каналів після налаштування QoS.

Висновки

Проаналізовано сучасні технології аналізу трафіку в комп’ютерних мережах, в результаті чого було доведено збільшення використання глибокому аналізі трафіку в комп’ютерних мережах.

Розглянуто технологію багато протокольної комутації по мітках, за допомогою якої реалізовано балансування трафіку мережі інтернет-сервіс провайдера , що дозволило зменшити смугу пропускання зовнішніх каналів. Також було досягнуто ефекту зменшення затримки проходження трафіку в мережі інтернет-сервіс провайдера.

Виконано метод балансування трафіку мережі інтернет сервіс провайдера з використанням технології глибокого аналізу трафіку, що дозволило збирати найрізноманітнішу статистику про трафік абонентів, захищати мережу від атак зловмисників , встановити кеш-сервер за допомогою якого досягнуто значну економію смуги пропускання зовнішніх каналів мережі інтернет сервіс провайдера

Аналіз результатів дослідження показав, що запропонований метод балансування трафіку може зменшити смугу пропускання зовнішніх каналів мережі інтернет-сервіс провайдера в години максимального навантаження і разом з тим забезпечити захист мережі від атак зловмисника, збір статистики без зменшення вимог до якості обслуговування.

Перелік використаних джерел

1.Гольдштейн А.Б. Технология и протоколы MPLS / А. Б. Гольдштейн, Б. С. Гольдштейн.— Санкт-Петербург «БХВ-Петербург», 2014 — 304с.— ISBN 5-8206-0126-2.

2. Vivek Alwayn. Advanced MPLS Design and Implementation / A. Vivek . Copenhagen University College of Engineering, 2004.— 480p. — ISBN 5-8459-0633-4.

3. Sniffer. [Електронний ресурс] / Режим доступу до журналу: https://www.opennet.ru/base/sec/arp_snif.txt.html

4. IANA Service Name and Transport Protocol Port Number Registry. [Електронний ресурс] /Режим доступу до журналу: http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml.

5. Економіка програмних і апаратних DPI на прикладі Cisco SCE і СКАТ. [Електронний ресурс] /Режим доступу до журналу: http://nag.ru /articles/article/28436/ekonomika-programmnyih-i-apparatnyih-dpi-na-primere-cisco-sce-i-skat.html

6. Платформи глибокого аналізу трафіку і управління трафіком додатків. [Електронний ресурс] /Режим доступу до журналу: http://www.inlinetelecom.ru /solutions/access_network/platform_depth_analysis_of_traffic_and_traffic_control_applications

7. "The Architecture of NGMON: Passive Network Monitoring System for High-Speed IP Networks", Accepted to appear in the Proc. of the 13th IFIP / IEEE International Workshop on Distributed Systems: Operations & Management (DSOM 2002)/ Se-Hee Han, Myung-Sup Kim, Hong-Taek Ju and James W. Hong — Montreal, Canada, October 21-23, 2002

8. InveaTech FlowMon.[Електронний ресурс] /Режим доступу до журналу: https://www.invea.com/en/products/flowmon

9. D Taylor, Survey and taxonomy of packet classification techniques/ Taylor D — ACM Comput. Surv.,2005 — pp. 238-275.

10.Cascarano N. Optimizing deep packet inspection for high-speed traffic analysis./N. Cascarano, L. Ciminiera F, Risso. — Network System Manager. 2011; — pp 7-31.

11. Olivain J. Detecting subverted cryptographic protocols by entropy checking./ J. Olivain J. Goubault-Larrecq. Research Report LSV-06-13, Laboratoire Specification et Verification, ENS Cachan, France, June 2006.

ДОДАТОК А

Міністерство освіти та науки України

Вінницький національний технічний університет

Інститут інформаційних технологій та комп'ютерної інженерії

ЗАТВЕРДЖУЮ

Завідувач кафедри _ОТ

д.т.н., професор Мартинюк Т. Б._____

(наук. ст., вч. зв., ініц. та прізв.) (підпис)

"____" _____20__р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання магістерської кваліфікаційної роботи

Метод балансування трафіку мережі Інтернет сервіс-провайдера з урахуванням вимог до якості обслуговування

08–23.МКР.004.00.000.ПЗ

Науковий керівник: к.т.н., доц. Захарченко С.М.

_____

(підпис)

студент групи _КІ-18м_

_____ Грабовський Є.В.

(підпис)

Вінниця 2019р.

1. Підстава для виконання магістерської кваліфікаційної роботи (МКР)

а) актуальність досліджень;

б) наказ про затвердження теми магістерської кваліфікаційної роботи.

2. Мета і призначення МКР

а) аналіз сучасних рішень по забезпеченню гарантованої якості обслуговування в мультисервісних телекомунікаційних мережах;

б) призначення розробки – виконання магістерської кваліфікаційної роботи, виконання організаційно – технологічних та наукових досліджень.

3. Вихідні дані для виконання МКР

- сучасні компютерні мережі;
- середовище розробки – компютеран мережі інтернет-сервіс провайдера.

4. Вимоги до виконання МКР

- огляд систем балансування трафіку;
- огляд систем аналізу трафіку;
- реалізувати теоретичний метод балансування трафіку мережі інтернет-сервіс провайдреа;

5. Етапи МКР та очікувані результати

№ з/п	Назва етапів роботи	Строк виконання етапів роботи	Примітк
1	Огляд технологій балансування трафіку , які використовуються в сучасних комп'ютерних мережах.		
2	Дослідження технологій аналізу трафіку та доцільність їхнього застосування в мережах різної складності.		
3	Розробка теоретичної моделі модернізації мережі інтернет-сервіс провайдера з урахуванням вимог до якості обслуговування		
4	Впровадження теоретичної моделі на мережі інтернет-сервіс провайдера		

5	Математична модель функціонування комп'ютерної системи в умовах невизначеностей оточення .		
6	Оформлення пояснювальної записки, графічного матеріалу і презентації		
7	Захист МКР		

6. Матеріали, що подаються до захисту МКР

Пояснювальна записка МКР, графічні і ілюстративні матеріали, протокол попереднього захисту МКР на кафедрі, відзив наукового керівника, відзив опонента, протоколи складання державних екзаменів, анотації до МКР українською та іноземною мовами, нормоконтроль про відповідність оформлення МКР діючим вимогам.

7. Порядок контролю виконання та захисту МКР

Виконання етапів графічної та розрахункової документації МКР контролюється науковим керівником згідно зі встановленими термінами. Захист МКР відбувається на засіданні Державної екзаменаційної комісії, затвердженою наказом ректора.

8. Вимоги до оформлення МКР

Вимоги викладені в МЕТОДИЧНИХ ВКАЗІВКАХ до дипломного проектування, ДСТУ_ 3008-95, ДСТУ 3974-2000 «Правила виконання дослідно-конструкторських робіт. Загальні положення» та діючого ГОСТ 2.114-95 ЕСКД.

9. Вимоги щодо технічного захисту інформації в МКР з обмеженим доступом

Відсутні.